

526,022

10/526022

(12)特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局(43) 国際公開日
2004年3月11日 (11.03.2004)

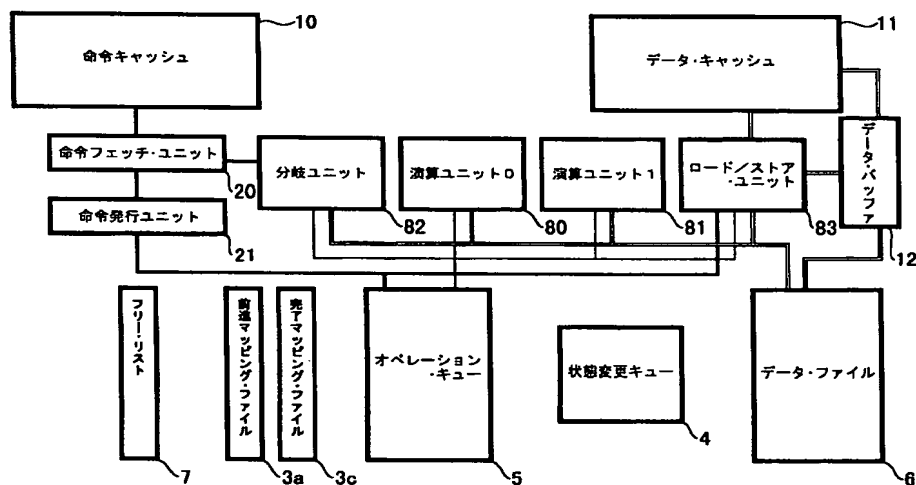
PCT

(10) 国際公開番号
WO 2004/021172 A1

- (51) 国際特許分類: G06F 9/38 (81) 指定国 (国内): AU, BR, CA, CN, CU, ID, IL, IN, JP, KR, MX, NO, NZ, PH, PL, SG, UA, US, VN, ZA.
- (21) 国際出願番号: PCT/JP2003/010891 (84) 指定国 (広域): ARIPO 特許 (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア特許 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ特許 (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI 特許 (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).
- (22) 国際出願日: 2003年8月27日 (27.08.2003)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ: 特願2002-254010 2002年8月30日 (30.08.2002) JP 添付公開書類: 国際調査報告書
- (71) 出願人 および (72) 発明者: 関一 (SEKI, Hajime) [JP/JP]; 〒790-0848 愛媛県松山市道後喜多町4番38号 Ehime (JP). 2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(54) Title: COMPUTER SYSTEM

(54) 発明の名称: 計算機システム



10...INSTRUCTION CACHE
11...DATA CACHE
20...INSTRUCTION FETCH UNIT
21...INSTRUCTION ISSUING UNIT
82...BRANCH UNIT
80...CALCULATION UNIT 0
81...CALCULATION UNIT 1
83...LOAD/STORE UNIT

12...DATA BUFFER
7...FREE LIST
3a...ADVANCE MAPPING FILE
3c...COMPLETION MAPPING FILE
5...OPERATION QUEUE
4...STATE MODIFICATION QUEUE
6...DATA FILE

(57) Abstract: In a computer system, each of operations which have been issued are executed in out-of-order. When they are normally completed, the corresponding part of the state holding means group is rewritten in response to this. After normal completion of all the operations included in the operation group combined with the advance mapping

[続葉有]

WO 2004/021172 A1



file content modification indicated by the head entry content of the state modification queue is indicated by the corresponding state holding means, the content of the completion mapping file is modified according to the head entry content of the state modification queue. The head entry is excluded from the state modification queue.

(57) 要約:

発行済のオペレーションの各々はout-of-orderで実行され、正常終了すると、これに応じて状態保持手段群の対応する部分が書き換えられるようになっており、状態変更キューの先頭のエントリの内容で示される前進マッピング・ファイルの内容変更と組み合わせさせたオペレーション群に含まれるオペレーションが全て正常終了したことが対応する状態保持手段によって示された後に、該状態変更キューの該先頭のエントリの該内容に基づき完了マッピング・ファイルの内容変更を行い、該状態変更キューから該先頭のエントリを除外するようになっている計算機システムである。

明 細 書

計算機システム

技術分野

本発明は、高度の命令レベル並列性を実現することにより高性能化の可能な
5 新規な構成の計算機方式に関するものである。

背景技術

従来、汎用の計算機として実用化されてきた計算機方式は、スタック・マシン
とレジスタ・ベースのマシンの2つに大きく分けることができる。スタック・マ
シンは、プログラムのコード・サイズが小さくてすみ、高速な割り込み処理やコ
ンテキスト・スイッチングに優れているが、高速化が困難であった。他方、レジ
10 スタ・ベースのマシンは比較的高速化が容易であるが、その他の面ではスタッ
ク・マシンに劣る点が多い。

スタック・マシンとレジスタ・ベースのマシンは、互いにプログラムの互換性
が無く、また上述のように、いずれかが絶対的に優れているというものではな
15 い。

本発明の目的は、スタック・マシンの命令形式ともレジスタ・ベースのマシン
の命令形式とも親和性があり、かつ、高性能化の可能な計算機方式を実現する
ことにある。

発明の開示

20 本発明による計算機方式は、基本的に out-of-order 実行の可能なスーパース
カラ・アーキテクチャと同様なものであるが、ビット長の長い命令フォーマット

を採用し、その中にオペレーション群の内容及び仮想のオペランド・スタックあるいは論理レジスタ群の状態変更の内容を記述するようになっている。

本発明による計算機システムは、各々のエントリにデータが書き込まれるようになっているデータ・ファイルと、各々のエントリにデータ・ファイルのエン
5 トリのアドレスが書き込まれるようになっている前進マッピング・ファイル及び完了マッピング・ファイルと、各々のエントリに個々の命令によるマッピング・ファイルの状態変更の内容が書き込まれるようになっている状態変更キューと、それぞれ状態変更キューの個々のエントリに対応して設けられたオペレーション群の実行状態を保持する状態保持手段群と、発行済のオペレーション
10 を out-of-order で処理する手段とを具備する。上記発行済のオペレーションを out-of-order で処理する手段は様々な構成のものが考えられるが、基本的には、発行済のオペレーションの内容を保持するバッファであるオペレーション・ウィンドウ、オペレーションを実行する機能ユニット群等によって実現される。

本発明の計算機システムは、1 サイクル当たり 1 命令を発行して、前進マッピング・ファイルの内容変更、その変更の内容の状態変更キューへの書き込み、及びオペレーション群の発行（オペレーション・ウィンドウへの書き込み）を行う
15 ようになっている。結果データを生ずるオペレーションの発行の場合、結果データを保持すべくデータ・ファイルのエントリが新たに割り付けられ、これがデスティネーション・レジスタとなる。また、発行される各オペレーションにおいて、ソース・レジスタとなるのは、アドレスが前進マッピング・ファイルをアクセスして得られるデータ・ファイルのエントリ、あるいは、同じ命令に含まれる
20 オペレーションにおいてデスティネーション・レジスタとして割り付けられるデータ・ファイルのエントリである。

発行済のオペレーションは各々 out-of-order でいずれかの機能ユニットによって実行され、正常終了すると、これに応じて上記状態保持手段群の対応する
25

部分が書き換えられるようになっている。

状態変更キューの先頭のエントリの内容と同じ命令の発行に基づくオペレーションが全て正常終了したことが対応する状態保持手段によって示された後に、状態変更キューの先頭のエントリの内容に基づき、その命令が発行された際の
5 前進マッピング・ファイルの内容変更を再現すべく、完了マッピング・ファイルの内容変更を行い、状態変更キューから上記先頭のエントリを除外するようになっている。

本発明による計算機システムは、スタック・モードとレジスタ・モードの2つのモードで動作することができる。スタック・ベースのプロセッサの機械語で記
10 述されたプログラムを本発明の計算機システムの命令形式に変換して実行する場合に、スタック・モードとなる。他方、RISCあるいはCISCのレジスタ・ベースのプロセッサの機械語で記述されたプログラムを本発明の計算機システムの命令形式に変換して実行する場合に、レジスタ・モードとなる。スタック・マシン
あるいはレジスタ・マシンの複数の命令を、本発明の計算機システムにおける1
15 つの命令に簡単に変換できるので、広範なプログラムを高速処理できる。

スタック・モードにおいては、マッピング・ファイルは、一種のスタックの内容を保持すべく用いられる。従来のスタックマシンにおいて、オペランド・スタックが ..., word3, word2, word1 (右端がスタックトップ) となっている状態
は、本発明による計算機システムにおいて、マッピング・ファイルに保持される
20 内容がエントリ・アドレス順に a, b, c, ... であれば、データ・ファイルにおいてエントリ・アドレスが a, b, c, ... である各エントリに、それぞれ word1, word2, word3, ... が保持されている状態に対応する。

レジスタ・モードにおいては、マッピング・ファイルは、レジスタ・ベースのスーパースカラ・プロセッサにおけるレジスタ・マッピング・テーブルと同等なものとして用いられる。
25

本発明は、同様の趣旨に基づく出願（国際出願番号 PCT/JP01/07442）において開示された発明に改良を加えたものである。その主な改良点は、①状態変更キューの個々のエントリに対応してオペレーション群の実行状態を保持する状態保持手段を設けたこと、及び②スタック・モードにおいてマッピング・ファイルがスタックの内容をその先頭からエントリ・アドレス順に保持するようにしたこと、である。

図面の簡単な説明

第1図は、本発明にかかる好ましい計算機システムの基本構成を示したブロック図である。

10 第2図は、本発明にかかる好ましい計算機システムの命令フォーマットを示した概念図である。

第3図は、スタック・モードにおける、前進マッピング・ファイル、完了マッピング・ファイル、及び後述する2つのボトム・ポインタの関係を示した説明図である。

15 第4図は、データ・ファイルのエントリの構成を示した説明図である。

第5図は、後述するオペレーション・キューのエントリの構成を示した説明図である。

第6図は、状態変更キューのエントリの構成を示した説明図である。

第7図は、フリー・リストの内部構成を示した説明図である。

20 第8図～第14図は、スタック・モードの一動作例における7時点の、前進マッピング・ファイル、完了マッピング・ファイル、状態変更キュー、オペレーション・キュー、データ・ファイル、及びフリー・リストの内容を具体的に示した説明図である。

第15図～第21図は、レジスタ・モードの一動作例における7時点の、前進

マッピング・ファイル、完了マッピング・ファイル、状態変更キュー、オペレーション・キュー、データ・ファイル、及びフリー・リストの内容を具体的に示した説明図である。

発明を実施するための最良の形態

5 以下に、本発明にかかる好ましい計算機システムについて説明する。

(1) 基本構成

まず、本実施例の計算機システムの基本的な構成について説明する。

第1図は計算機システムのブロック図であって、10は命令キャッシュ、11はデータ・キャッシュ、12はデータ・バッファ、20は命令フェッチ・ユニット、21は命令発行ユニット、3aは前進マッピング・ファイル、3cは完了マ
10 ッピング・ファイル、4は状態変更キュー、5はオペレーション・キュー、6はデータ・ファイル、7はフリー・リスト、80及び81は各々演算ユニット0及び1、82は分岐ユニット、83はロード／ストア・ユニットを表している。

本発明の計算機システムにおいて状態変更キューの個々のエントリに対応し
15 て設ける必要があるオペレーション群の実行状態を保持する状態保持手段は、本実施例においては、後述する4つのオペレーション・ステータス・フィールドによって実現している。

本発明の計算機システムは、発行済のオペレーションの内容を保持するバッ
ファであるオペレーション・ウインドウを具備する必要がある。オペレーシ
20 ョン・ウインドウを実現するには、基本的に集中型と分散型の2通りの方法があるが、本実施例の計算機システムは集中型のオペレーション・キューを備えるものとしている。

本発明による計算機システムは、スタック・モードとレジスタ・モードの2つのモードで動作することができる。スタック・ベースのプロセッサの機械語で記

述されたプログラムを本発明の計算機システムの命令形式に変換して実行する場合に、スタック・モードとなる。他方、RISC あるいは CISC のレジスタ・ベースのプロセッサの機械語で記述されたプログラムを本発明の計算機システムの命令形式に変換して実行する場合に、レジスタ・モードとなる。

- 5 本発明の計算機システムの具備するマッピング・ファイルは、各々のエントリにデータ・ファイルのエントリのアドレスが書き込まれるような構成となっており、レジスタ・モードにおいては、レジスタ・ベースの out-of-order 実行の可能なスーパースカラ・プロセッサにおけるレジスタ・マッピング・テーブルと同様に機能する。本発明において、マッピング・ファイルは、スタック・モードに
10 おいても同等な機能を有するものである。

(2) 命令形式

- 第2図は本発明実施例の計算機システムの命令フォーマットを示す概念図であって、スタック・モードにおいてもレジスタ・モードにおいても、個々の命令のフォーマットは、基本的に、フォーマット情報フィールド101、オペレー
15 ション・フィールド102及び状態変更フィールド103からなる。

フォーマット情報フィールド (FI フィールド) では、当該命令の長さや、その命令が含むオペレーションの数などに関する情報が示される。

- オペレーション・フィールド (Op フィールド) には、算術論理演算や分岐、ロード／ストアなどからなるオペレーション群の内容が示される。本実施例にお
20 いては、個々のオペレーションは、RISC プロセッサにおける命令と同様な形式で記述される。本実施例においては、1つの命令は最大4つまでのオペレーションを含むものとする (分岐オペレーションが含まれるような場合には、あえて1命令に4つのオペレーションを含める必要はない)。

状態変更フィールド (SM フィールド) では、マッピング・ファイルの内容の変

更について示される。即ち、スタック・モードにおいては仮想のオペランド・スタック、レジスタ・モードにおいては仮想の論理レジスタ群の状態変更の内容が示される。後述するように、状態変更フィールド内部のフォーマットは、スタック・モードとレジスタ・モードで異なる。

- 5 結果データを生ずるオペレーションの発行の場合、結果データを保持すべくデータ・ファイルのエントリが割り付けられるが、そのエントリの論理的な位置づけは、SM フィールドの内容によって示されることになる。

本実施例の計算機システムは、1 サイクル当たり上記の形式の1 命令を発行できるものである。

10 (2-1) スタック・モードにおける命令形式

以下に、スタック・モードにおける命令形式について説明する。

- 本実施例の計算機システムにおけるデータ・ファイルの各エントリが、仮想のスタック・マシンのオペランド・スタックの1 エントリの保持するデータを保持できるものとする。本実施例においては、マッピング・ファイルのアドレス 0, 1, 15 2, ... のエントリの内容をそれぞれ s_0 , s_1 , s_2 , ... とおくとき、エントリ・アドレスが s_0 , s_1 , s_2 , ... であるデータ・ファイルの各エントリにそれぞれ保持されるべきデータを順に並べることによって、仮想のスタック・マシンのオペランド・スタックの内容 (s_0 がスタックトップに対応) が構成されるべきものとする。

- 20 例えば、 $A \cdot X + B / X$ を計算し、その結果を変数 Y としてメモリにストアするプログラムは、スタック・マシンにおいては、次のように書ける。

LOAD <4>	; ローカル変数 4 のデータをオペランド・スタックにプッシュする
DUP	; スタックトップのデータをコピーし、そのコピーをスタックに積む
LOAD <1>	; ローカル変数 1 のデータをオペランド・スタックにプッシュする
MUL	; 2 変数分のデータをポップして乗算を行い、結果をプッシュする
SWAP	; オペランド・スタックにおいて最上位とその次のデータを入れ替える
LOAD <2>	; ローカル変数 2 のデータをオペランド・スタックにプッシュする
SWAP	; オペランド・スタックにおいて最上位とその次のデータを入れ替える
DIV	; 2 変数分のデータをポップして除算を行い、結果をプッシュする
ADD	; 2 変数分のデータをポップして加算を行い、結果をプッシュする
STORE <5>	; スタックトップのデータをローカル変数 5 にストアする

ここで、変数 A, B, X, Y の格納域を、それぞれローカル変数 1, 2, 4, 5 としている。

上記プログラムがどのように本実施例の計算機システムにおけるスタック・モードの命令に変換されるかを説明する。

- 5 上記プログラムを 2 つの命令群に分割して、個々の命令について、オペレーションの内容とマッピング・ファイルに加えられる変更の内容を分けて書くと、次のようになる。

命令 1a	load f1, <4>	; +1: f1
命令 1b	add f2, f1, 0	; +2: f1, f2
命令 1c	load f3, <1>	; +3: f1, f2, f3
命令 1d	mul f4, f2, f3	; +2: f1, f4
命令 1e	(swap)	; +2: f4, f1
命令 2a	load f1, <2>	; +1: f1
命令 2b	(swap)	; +1: f1, s0
命令 2c	div f2, f1, s0	; 0: f2
命令 2d	add f3, s1, f2	; -1: f3
命令 2e	store <5>, f3	; -2:

ここで、結果データを保持すべく新たに割り付けられるデータ・ファイルのエントリのアドレスを、各々の命令群において順に f1, f2, ... とおいている。

本実施例においては、f1, f2, ... の各々は、各サイクルにおいてフリー・リストから取り出されるものに対応する。

また、マッピング・ファイルに加えられる変更に関しては、スタックが何エン
トリ分成長するかという量と、スタックトップ近傍に加えられる変更内容（右
5 端がスタックトップに対応）を示している。

命令 1a の意味は、ローカル変数<4>のデータをロードし f1 に対応するデータ・
ファイルのエントリに書き込み、その要素がマッピング・ファイルに保持される
スタックを 1 エントリ分成長させ、新たにスタックトップとなるエントリの内
容を f1 に対応するデータ・ファイルのエントリのアドレスとする、というもの
10 である。命令 1b の意味は、f1 に対応するデータ・ファイルのエントリのデータ
に 0 を加え、その結果データを f2 に対応するデータ・ファイルのエントリに書
き込み、その要素がマッピング・ファイルに保持されるスタックを 1 エントリ分
成長させ、新たにスタックトップとなるエントリの内容を f2 に対応するデータ・
ファイルのエントリのアドレスとする、というものであるが、マッピング・ファ
15 イルに加えられる変更については、命令 1a による変更内容も含めて記述してい
る。その他の命令も同様である。

2 つの命令群の各々について、その内容をまとめて、オペレーションとマッ
ピング・ファイルの内容の変更に分けて記述すれば次のようになる。

Op{ load f1, <4>; add f2, f1, 0; load f3, <1>; mul f4, f2, f3 }
SM{ +2: f4, f1 }
Op{ load f1, <2>; div f2, f1, s0; add f3, s1, f2; store <5>, f3 }
SM{ -2: }

これが、上記プログラムを本発明実施例の計算機システムのスタック・モード
20 の命令形式に変換した場合の、2 命令の各々の Op フィールドと SM フィールド
の内容である。このように、スタック・モードの命令形式では、SM フィールドに

において、スタックの成長量と共に、更新後のスタックトップ近傍の内容を表す符号の系列が示されるようになっている。

(2-2) レジスタ・モードにおける命令形式

次に、レジスタ・モードにおける命令形式について説明する。

- 5 本実施例の計算機システムにおけるデータ・ファイルの各エントリが、仮想のレジスタ・マシンの1レジスタの保持するデータを保持できるものとする。本実施例においては、マッピング・ファイルのアドレス 0, 1, 2, ... のエントリの内容をそれぞれ r0, r1, r2, ... とおくとき、エントリ・アドレスが r0, r1, r2, ... であるデータ・ファイルの各エントリに、それぞれ仮想のレジスタ・マシンのレジスタ R0, R1, R2, ... のデータが保持されるべきものとする。
- 10

例えば、メモリから変数 A, B, X のデータをレジスタ群にロードして、 $(A \times X + B/X) \times 2$ を計算し、その結果を変数 Y としてメモリにストアするプログラムは、レジスタ・マシンにおいては、次のように書ける。

LOAD	R1, 100[R0]	; (R0 の中身+100)番地のデータをレジスタ R1 にロードする
LOAD	R4, 400[R0]	; (R0 の中身+400)番地のデータをレジスタ R4 にロードする
MUL	R5, R1, R4	; $R5 = R1 \times R4$
LOAD	R2, 200[R0]	; (R0 の中身+200)番地のデータをレジスタ R2 にロードする
DIV	R4, R2, R4	; $R4 = R2/R4$
ADD	R5, R5, R4	; $R5 = R5 + R4$
MUL	R5, R5, 2	; $R5 = R5 \times 2$
STORE	500[R0], R5	; レジスタ R5 の内容を (R0 の中身+500)番地にストアする

- ここで、変数 A, B, X, Y のメモリにおける格納域のアドレスを、R0 の中身にそれぞれ 100, 200, 400, 500 を加えたものとしている。
- 15

上記プログラムがどのように本実施例の計算機システムにおけるレジスタ・モードの命令に変換されるかを説明する。

上記プログラムを4命令ずつから成るグループに分割して、個々の命令について、オペレーションの内容とマッピング・ファイルに加えられる変更の内容を分けて書くと、次のようになる。

命令 1a	load f1, 100[r0] ; 1:f1
命令 1b	load f2, 400[r0] ; 4:f2
命令 1c	mul f3, f1, f2 ; 5:f3
命令 1d	load f4, 200[r0] ; 2:f4
命令 2a	div f1, r2, r4 ; 4:f1
命令 2b	add f2, r5, f1 ; 5:f2
命令 2c	mul f3, f2, 2 ; 5:f3
命令 2d	store 500[r0], f3

5

ここで、スタック・モードの場合と同様に、結果データを保持すべく新たに割り付けられるデータ・ファイルのエントリのアドレスを、各々の命令群において順に f1, f2, ... とおいている。

命令 1a の意味は、(r0 に対応するデータ・ファイルのエントリのデータ+100) 番地のデータをロードし f1 に対応するデータ・ファイルのエントリに書き込み、マッピング・ファイルのアドレス 1 のエントリの内容を f1 に対応するデータ・ファイルのエントリのアドレスに置き換える、というものである。命令 1c の意味は、f1 及び f2 に各々対応するデータ・ファイルのエントリのデータを掛け合わせ、その結果データを f3 に対応するデータ・ファイルのエントリに書き込み、マッピング・ファイルのアドレス 5 のエントリの内容を f3 に対応するデータ・ファイルのエントリのアドレスに置き換える、というものである。その他の命令も同様である。

10

15

2つの命令群の各々について、4命令の内容をまとめて、オペレーションとマッピング・ファイルの内容の変更に分けて記述すれば次のようになる。

Op{load f1, 100[r0]; load f2, 400[r0]; mul f3, f1, f2; load f4, 200[r0]}
SM{1:f1, 2:f4, 4:f2, 5:f3}
Op{div f1, r2, r4; add f2, r5, f1; mul f3, f2, 2; store 500[r0], f3}
SM{4:f1, 5:f3}

これが、上記プログラムを本発明実施例の計算機システムのレジスタ・モードの命令形式に変換した場合の、2命令の各々の Op フィールドと SM フィールドの内容である。この2番目の命令に変換する際に、マッピング・ファイルのアドレス5のエントリの内容の置き換えが2つあるが、SM フィールドの内容としては最後のものだけが残されている。このように、レジスタ・モードの命令形式では、SM フィールドにおいて、内容を変更すべきマッピング・ファイルのエントリ
5 のアドレスと個々の変更内容を示す符号の組が列挙されるようになっている。

V L I W (very long instruction word) アーキテクチャに基づく個々の命令も、容易に本発明の計算機システムのレジスタ・モードの命令形式に変換することができる。

(3) スタック・モード

(3-1) スタック・モードにおける動作に必要な機能と構成

10 ここでは、本実施例の計算機システムの各構成要素ごとに、スタック・モードの動作において利用される機能と構成について説明する。これらは、一部を除き、レジスタ・モードにおいても必要とされる機能と構成である。

(A) 命令フェッチ・ユニット

15 命令フェッチ・ユニットは、図示していないプログラムカウンタを具備しており、命令キャッシュから命令をフェッチし、命令発行ユニットに渡すようになっている。分岐の予測や分岐の実行も担う。

(B) 命令発行ユニット

命令発行ユニットは、命令フェッチ・ユニットから渡された命令の発行のために、その内容に基づき前進マッピング・ファイルやデータ・ファイルの操作、オペレーション・キューと状態変更キューへの書き込みを行うための各種信号を発生するようになっている。

(C) マッピング・ファイル

マッピング・ファイルは、各々のエントリにデータ・ファイルのエントリのアドレスが書き込まれるような構成となっている。

本発明実施例の計算機システムのスタック・モードにおいては、マッピング・ファイル及びデータ・ファイルによって、スタック・マシンに具備されるべきスタックの最上位の部分が仮想的に構成されるようになっている。即ち、スタック・モードにおいては、マッピング・ファイルは、一種のスタックの内容を保持すべく用いられる。従来のスタックマシンにおいて、オペランド・スタックが ..., word3, word2, word1 (右端がスタックトップ) となっている状態は、本発明による計算機システムにおいて、マッピング・ファイルのアドレス 0, 1, 2, ... のエントリの内容がそれぞれ a, b, c, ... であれば、データ・ファイルにおいてエントリ・アドレスが a, b, c, ... である各エントリに、それぞれ word1, word2, word3, ... が保持されている状態に対応する。オペランド・スタックの規模が大きくなると、残りの部分はデータ・バッファさらにはデータ・キャッシュに格納されるようになっている。

本発明の計算機システムは、前進マッピング・ファイル (AMF ; Advanced Mapping File) と完了マッピング・ファイル (CMF ; Completed Mapping File) の2つの同形のマッピング・ファイルを具備する。

本発明の計算機システムにおいては、命令が発行されるごとに、その命令の内容に応じてAMFの内容を変更する。すなわち、AMFは発行済みの全ての

命令の内容を反映している。

他方、CMFは、プログラム上の順番で完了済みの全ての命令の内容を反映するものである。本発明の計算機システムは発行済みの命令に含まれるオペレーションの out-of-order 実行を可能とするものであるが、CMFは、正確な例外処理を保証するため、in-order で完了済の全ての命令に基づく状態を構成するために存在するものである。

各マッピング・ファイルごとにスタックの要素を保持する最下位のエントリを示すボトム・ポインタと呼ぶレジスタが存在する。初期状態あるいはマッピング・ファイルが空の場合には、ボトム・ポインタの値は-1であるものとする。

第3図は、本実施例の計算機システムにおける、2組のマッピング・ファイルとボトム・ポインタの関係を示す説明図である。2つのマッピング・ファイルAMF 3 a及びCMF 3 cは同数のエントリを有し、各マッピング・ファイルで各々のエントリに上から順に0, 1, 2, ... とアドレスが付けられている。斜線が施されていないAMF/CMFのエントリは、スタックの構成要素としてデータ・ファイルのエントリのアドレスを保持しているものとする。第3図に示すように、AMF及びCMFの各々に対して設けられたボトム・ポインタを、それぞれB a及びB cと名付けている。

(D) データ・ファイル (DF ; Data File)

データ・ファイル (DF) は、各々のエントリに1語分のデータが書き込まれるような構成となっている。

本実施例においては、DFのエントリにはp00, p01, p02, ... のようにアドレスが付けられているものとする。

第4図は、本実施例の計算機システムにおける、DF 6の各々のエントリ6 (i)の構成を示す説明図である。ここで、iはエントリのアドレスである。DF 6の各々のエントリ6 (i)はデータ・フィールド6 1 (i)、及び書込フラグ (W

F ; Write Flag) フィールド 6 2 (i) から成っている。

実際の DF のハードウェア上の構成は、上述の各フィールド別に設けられたレジスタ・ファイルの集合体である。

DF の各々のエントリのデータ・フィールドは、1 語分のデータが書き込まれるようになっている。
5

DF の各々のエントリにおいて、WF フィールドは、データ・フィールドにデータの書き込みが完了していれば 1、完了していなければ 0 が書き込まれているようになっている。

(E) オペレーション・キュー (OQ ; Operation Queue)

10 オペレーション・キュー (OQ) は、未完了の発行済命令に含まれるオペレーションの内容を保持するバッファであり、循環型の FIFO キューの構成となっている。

第 5 図は、本実施例の計算機システムにおける、OQ 5 の各々のエントリ 5 (i) の構成を示す説明図である。ここで、i はエントリのアドレスである。OQ 5 の各々のエントリ 5 (i) はオペレーション・フィールド 5 0 (i)、デスティネーション・フィールド 5 1 (i)、オペランド・フィールド 5 2 (i)、第 1 ソース・フィールド 5 3 (i)、第 1 ソース書込フラグ (SWF 1) フィールド 5 4 (i)、第 2 ソース・フィールド 5 5 (i)、第 2 ソース書込フラグ (SWF 2) フィールド 5 6 (i)、リポート・フィールド 5 7 (i)、及びディスパッチ・フラグ・フィールド 5 8 (i) から成っている。
15
20

OQ の各々のエントリのオペレーション・フィールドはオペレーション・コードが書き込まれるようになっている。

OQ の各々のエントリのデスティネーション・フィールドは、結果データを生ずるオペレーションの場合に、その結果データを保持すべく割り付けられる DF のエントリのアドレスが書き込まれるようになっている。
25

○Qの各々のエントリのオペランド・フィールドは、命令の 0p フィールド中にオペランド値が示されるようなオペレーションの場合に、そのオペランド値が書き込まれるようになっている。

○Qの各々のエントリの第1、第2ソース・フィールドの各々は、オペレーションのソース・データを保持すべく割り付けられているDFのエントリのアドレスが書き込まれるようになっている。

○Qの各々のエントリにおいて、第1、第2の各SWFフィールドは各々第1、第2ソース・フィールドに対応して設けられている。SWF 1/2フィールドは第1/2ソース・フィールドに示されるDFのエントリにデータの書き込みが完了していれば1、完了していなければ0が書き込まれているようになっている。

○Qの各々のエントリのリポート・フィールドは、そのオペレーションを含む命令に対応する状態変更キューのエントリのアドレスと識別番号(A, B, C, Dのうちのいずれか)が書き込まれるようになっている。

○Qの各々のエントリにおいて、ディスパッチ・フラグ・フィールドは、オペレーションがまだディスパッチされていなければ0、ディスパッチされていれば1が書き込まれているようになっている。

(F) 状態変更キュー (SMQ ; State Modification Queue)

状態変更キュー (SMQ) は、未完了の発行済命令の各々によるマッピング・ファイルの状態変更の内容を保持するバッファであり、循環型のFIFOキューの構成となっている。本実施例においては、SMQの各々のエントリは1つの命令に対応するものとなっている。

第6図は、本実施例の計算機システムにおける、SMQ 4の各々のエントリ 4(i)の構成を示す説明図である。ここで、iはエントリのアドレスである。SMQ 4の各々のエントリ 4(i)は状態変更 (SM) フィールド 40(i)、オペレ

ーション・ステータスA (OS__A) フィールド4 1 (i)、オペレーション・ステータスB (OS__B) フィールド4 2 (i)、オペレーション・ステータスC (OS__C) フィールド4 3 (i)、及びオペレーション・ステータスD (OS__D) フィールド4 4 (i)から成っている。

- 5 実際のSMQのハードウェア上の構成は、上述の各フィールド別に設けられたレジスタ・ファイルの集合体である。

SMQの各々のエントリのSMフィールドは、対応する命令のSMフィールドに示されたマッピング・ファイルの変更内容が書き込まれるようになっている。

- 10 SMQの各々のエントリにおいて、OS__A、OS__B、OS__C、OS__Dの各フィールドは、各々対応するオペレーションの実行状態に関する情報が書き込まれているようになっている。本実施例においては、簡単のため、対応するオペレーションが存在しないか正常終了していれば1、その他の場合には0が書き込まれているものとする。

15 (G) データ・バッファ

データ・バッファは、各々のエントリに1語のデータが書き込まれるようになっている循環型のバッファである。

(H) 機能ユニット

- 20 本実施例の計算機システムは、第1図に示すように、演算ユニット0及び1、分岐ユニット及びロード／ストア・ユニット (LSU ; Load Store Unit) の4つの機能ユニットを具備するものとしている。これらは、各々パイプライン処理が可能で、互いに独立に並行して動作するものである。

スタック・モードにおいては、LSUはデータ・バッファ及びデータ・キャッシュにアクセスすることができるようになっている。

- 25 本発明実施例の計算機システムのスタック・モードにおいては、マッピング・

ファイル及びデータ・ファイル（DF）によって、スタック・マシンに具備されるべきスタックの最上位の部分が仮想的に構成されるが、その下の部分がデータ・バッファ、さらにその下の部分がデータ・キャッシュに格納されるようになっている。LSUはデータ・バッファに高速にアクセスできるようになっており、

5 アクセスすべき変数データがデータ・バッファに保持されている割合が大きいほど、より効率的な計算が可能となる。また、データ・バッファに適当な語数のデータを溜めておくようにすることによって、後述するDFーデータ・バッファーデータ・キャッシュの間の Spill/Fill の動作を効率的に行うことができる。

LSUは、最初のローカル変数へのポインタを保持する図示してないレジスタ（vars レジスタ）を具備する。本実施例の計算機システムにおいては、ローカル変数の格納域はデータ・バッファあるいはデータ・キャッシュにあるが、vars レジスタには、データ・キャッシュにおける相当するアドレス値が書き込まれているようになっている。つまり、全てあるいは一部のローカル変数のデータが実際にはデータ・バッファに保持されていても、各々のローカル変数に、

15 全てのローカル変数をデータ・キャッシュに Spill したと仮定した場合のデータ・キャッシュにおけるアドレス値を対応させることができる。ロード／ストア・オペレーションの処理において、LSUは vars レジスタの値を用いてアドレス計算を行い、対象となるローカル変数の格納域がデータ・バッファかデータ・キャッシュかを判定し、その格納域にアクセスする。

20 LSUは、プログラム中に示されるロード／ストア・オペレーションを実行すると共に、オーバーフロー／アンダーフローの回避のため、AMF／CMF及びDFによって構成される仮想の部分スタックの最下位にあたるデータをデータ・バッファとの間で自動的に Spill/Fill するようになっている。

1語分のデータをDFからデータ・バッファに Spill するには、(a) AMFと

25 CMFで各々のボトム・ポインタ（Ba、Bc）で示されるエントリの内容が一

致している、(b)その一致する内容と同じDFのエントリのアドレスがいずれかのソース・フィールドに書き込まれたエントリがOQに保持されるキュー内に存在しない、という2つの条件が満たされていなくてはならない(そうでない場合は満たされるまで待つ)。条件(b)に関しては、命令に含まれるオペレーションのソース・レジスタとして用いられるDFのエントリのアドレスがマッピング・ファイル中に保持された状態で残るケースに一定の制約を設けることによって、常に満たされるようにすることもできる。命令体系にそのような制約を設けない場合には、計算機システムは、OQにおいて各ソース・フィールドで上記一致する内容であるDFのエントリのアドレスを照合し、上記条件(b)に関するチェックを行うような機能を備えた構成としなければならない。上記2つの条件が満たされる場合、AMF/CMFのボトム・ポインタBa/Bcで示されるエントリの内容で示されるDFのエントリに書き込まれている1語分のデータをデータ・バッファにSpillすることができる。その際、Ba及びBcの値から1を引き、上記DFのエントリのアドレスをFLに登録する。

逆に、データ・バッファからDFに1語分のデータをFillするには、データ・バッファからFillすべき1語分のデータを取り出し、それにフリーなDFの1エントリを割り付け、そのデータ・フィールドに書き込む。WCFフィールドは1とする。さらに、ボトム・ポインタBa及びBcの値に1を加え、上記割り付けられるDFのエントリのアドレスを、AMF及びCMFの各々1を加えたボトム・ポインタ値で示されるエントリに各々書き込む。

データ・バッファとデータ・キャッシュの間でも、データ・バッファの空きに応じて適宜Spill/Fillの動作が行われるようになっている。

以上では、命令の発行/完了に伴うAMF/CMFの操作がないものとして、DFとデータ・バッファの間のSpill/Fillの動作について説明したが、命令の発行/完了とSpill/Fillの動作を合成し、同時に行えるような構成は容易に実

現できる。

また、本発明の計算機システムは、DFとデータ・バッファの間及びデータ・バッファとデータ・キャッシュの間で、一度に複数語のデータを Spill/Fill できるような構成とすることも可能である

5 (I) フリー・リスト (FL ; Free List)

フリー・リスト (FL) は、フリーな、即ち、割り付けられていないDFのエントリのアドレスを保持するもので、本実施例においては、第7図に示すように、命令完了用登録バッファ (EBIC ; Entry Buffer for Instruction Completion) 71a、スタック・スピル用登録バッファ (EBSS ; Entry Buffer for Stack Spill) 71b、デスティネーション用割付キュー (AQD ; Allocation Queue for Destinations) 72a、及びスタック・フィル用割付キュー (AQSF ; Allocation Queue for Stack Fill) 72bを備える。これらは各々適当な数のDFのエントリのアドレスを保持でき、AQDとAQSFは循環型のFIFOキューの構成となっている。

15 初期状態においては、DFの各々のエントリのアドレスはAQDとAQSFのどちらかに保持されている。新たにDFのエントリを割り付ける必要がある場合にFLからフリーなDFのエントリのアドレスが取り出されるが、命令の発行に伴いオペレーションの結果データを保持すべく割り付けるためにはAQDから、また、その要素がマッピング・ファイルに保持されるスタックの Fill
20 のためにはAQSFから取り出されるようになっている。逆に、割り付けが解除されるDFの各々のエントリについてはそのアドレスがFLに登録されるようになっているが、命令の完了に伴い割り付けが解除されるものはEBICに、また、その要素がマッピング・ファイルに保持されるスタックの Spill に伴い割り付けが解除されるものはEBSSに書き込まれるようになっている。EBIC
25 C/EBSSに書き込まれた各々のDFのエントリのアドレスは、すぐに、A

QD及びAQSFの保持する内容量に応じて、そのいずれかに移されるようになっている。

(3-2) スタック・モードにおける動作

5 ついで、本発明実施例の計算機システムのスタック・モードにおける動作を説明する。

本実施例の計算機システムは、基本的に、命令を、①命令フェッチ、②命令発行、③オペレーション実行、④命令完了の4段階で処理する。以下に各段階ごとに動作内容を説明する。

① 命令フェッチ

10 この段階では、命令フェッチ・ユニットが命令キャッシュから命令を取り出すと共に、次にフェッチする命令のアドレスを決定する。次に命令をフェッチするのは通常次アドレス値からであるが、フェッチした命令が無条件分岐オペレーションを含むか、条件分岐オペレーションを含み分岐すると予測した場合、分岐予測が外れた場合、あるいは例外が発生した場合には、フェッチするアド
15 レス値を変更する。

② 命令発行

この段階では、発行する命令のOpフィールドの内容に基づくオペレーション群の発行（オペレーション・キュー（OQ）への書き込み）、SMフィールドの内容に基づく前進マッピング・ファイル（AMF）及びそのボトム・ポインタBa
20 の内容変更、及びその変更の内容の状態変更キュー（SMQ）への書き込みが行われる。この際、命令中のs0, s1, s2, ... の各々は、スタックの最上位、2番目、3番目、... に対応するが、それぞれ変更前のAMFのアドレス0, 1, 2, ... のエントリの内容に、f1, f2, ... の各々はそれぞれフリー・リスト（FL）から順に取り出されるDFのエントリの内容に、置き換えられる。

AMF及びB aの内容変更に関しては、次のように行われる。まず、命令のSMフィールドに示される各符号が上記の要領でそれぞれ適切なDFのエントリのアドレスに置き換えられ、それぞれ対応するAMFのエントリに書き込まれる。ここで、AMFのアドレス0のエントリがスタックトップ即ち命令のSMフィールドにおいて右端に示される符号に対応し、以下アドレス順にAMFのエントリとスタックの要素が対応する。命令のSMフィールドで変更内容が明示されない部分については、スタックの成長量だけエントリの内容がシフトすることになる。即ち、スタックの成長量を g とおくと、AMFのアドレス i の内容が命令発行によりAMFのアドレス $(i+g)$ に書き込まれることになる。

5 また、ボトム・ポインタB aの値にはスタックの成長量を加えられる。

発行される命令に含まれるオペレーションの内容が書き込まれるOQの各エントリにおいて、リポート・フィールドには、同じ命令に基づく書き込みが行われるSMQのエントリのアドレスと識別番号（順にA, B, C, Dと機械的に付すものとする）が書き込まれる。OQのエントリのディスパッチ・フラグ・フィールドには0が書き込まれる。

15

命令の発行に伴い（ $f1, f2, f3, \dots$ に対応する）新たに割り付けられるDFのエントリにおいては、WFフィールドに0が書き込まれる。

ストア・オペレーションの場合には、その内容が、OQに書き込まれると共に、LSUに送られる。

OQの書き込みが行われるソース・フィールドに対応するSWFフィールドはまず0とされるが、 $s0, s1, s2, \dots$ のいずれかから置き換えられたDFのエントリ・アドレスのソース・フィールドへの書き込みの各々に関しては、次サイクルにおいて、そのアドレスのDFのエントリのWFフィールドの内容が読み出され、これが1であれば、OQの対応するSWFフィールドが1に変更されるようになっている。

20

25

発行される命令に基づき書き込みが行われる SMQ のエントリにおいて、対応するオペレーションの存在するオペレーション・ステータス・フィールドには未実行を意味する 0 が書き込まれ、その他には 1 が書き込まれる。

③ オペレーション実行

- 5 OQ に保持される未実行のオペレーションは、実行可能となったものから、適切な機能ユニットにディスパッチされ、処理されるようになっている。従って、オペレーションの実行順序は out-of-order になる。

- 10 OQ において、オペレーションに必要なソース・データが全て DF の該当するエントリに書き込み済であることが対応する SWF フィールドの内容によって確認されるエントリがあれば、その保持するオペレーションの実行のため、そのエントリの内容が DF をアクセスして得られるソース・データと共に適切な機能ユニットに送られるようになっている。この際、その OQ のエントリのディスパッチ・フラグ・フィールドが 1 に変更される。

- 15 ロード・オペレーションや算術論理演算等の結果データを生じるオペレーションに関しては、結果データが正常に得られれば、デスティネーションである DF のエントリのデータ・フィールドに結果データが書き込まれ、WF フィールドが 1 に変更される。また、OQ において各ソース・フィールドで上記デスティネーションである DF のエントリのアドレスが照合され、一致するソース・フィールドに対応する SWF フィールドが 1 に変更される。

- 20 ストア・オペレーションを書き込みの内容とする OQ のエントリに関しては、同じ内容が命令発行段階において LSU に送られている。命令発行段階において確定しなかったアドレス計算に必要なソース・データは、本実施例においては、その DF への書き込みが OQ において確認された直後に、LSU に送られるようになっているものとする。

- 25 いずれのオペレーションも、その実行が正常に終了すれば、レポート・フィー

ルドの内容に基づき、対応するSMQのエントリのオペレーション・ステータス・フィールドが1に変更される。

5 ストア・オペレーションに関しては、アドレス計算はオペレーション実行段階において out-of-order で実行されるが、正確な例外処理を保証するために、ストアの実行は命令完了段階において行われる。従って、ストア・オペレーションの場合、ストア・データとストア・アドレスが共に確定すれば、対応するSMQのエントリのオペレーション・ステータス・フィールドが1に変更されるようになっている。

10 あるオペレーションの処理において例外事象が発生した場合には、その情報が、対応するSMQのエントリのオペレーション・ステータス・フィールドに書き込まれると共に、命令フェッチ・ユニットに通知される。

④ 命令完了

命令の完了はプログラム上の順番で行われる。

15 SMQのキューの先頭のエントリにおいて、全てのオペレーション・ステータス・フィールドの内容が1である、あるいはそうになると、対応する命令の完了が可能となる。命令の完了を実行するには、SMQの先頭のエントリの内容に基づきCMFおよびB cの内容を変更し、SMQにおいて上記先頭のエントリをキューから除外する。ここで、命令の発行の際にAMFに対して行われた内容変更がCMFにおいて再現されている。

20 また、命令の完了に伴い、割り付けを解除すべきDFのエントリのアドレスがそれぞれFLに登録される。スタック・モードにおいては、割り付けを解除すべきDFのエントリ群のアドレスは次の2つのグループからなる。命令の完了に伴う内容変更によりCMFにおいて保持されなくなるもの、及び、命令に含まれるオペレーションを保持していたOQのエントリ群のデスティネーション・フィールドに書き込まれていてCMFへの書き込みが行われないもの、であ

25

る。

ストア・オペレーションを含む命令を完了する場合には、LSUにストアの実行を依頼する。こうすれば、データがプログラム上の順番でストアされることが保証できる。

- 5 以上が、本発明実施例の計算機システムのスタック・モードにおける動作の概要である。

(3-3) スタック・モードにおける動作例

ついで、具体的な動作例について説明する。いま、本実施例の計算機システムで、前述の $Y=A \cdot X+B/X$ を計算する次の2命令からなるプログラムを実行する

10 ものとする。

命令1	Op{ load f1, <4>; add f2, f1, 0; load f3, <1>; mul f4, f2, f3 } SM{ +2: f4, f1 }
命令2	Op{ load f1, <2>; div f2, f1, s0; add f3, s1, f2; store <5>, f3 } SM{ -2: }

- 第8図～第14図は、本実施例の計算機システムにおいて、上記プログラムを処理する際の状態の変化を時系列に示した説明図であり、以下ではこの図をもとに詳細な動作を説明する。第8図～第14図において、DF6、OQ5及びSMQ4の各エントリの構成は、それぞれ第4図、第5図及び第6図のものと同じである。マッピング・ファイルのボトム・ポインタが示すエントリより下のエントリや、FIFOキューの構成となっている構成要素のキューから外れるエントリなど、書込み内容が意味あるものとして保持されていないような部分には、斜線が施されている。p##が記入されている箇所は、DFのいずれかのエントリのアドレスが書き込まれているが、本動作例の説明において留意する
- 15
- 20 必要がないことを意味する。また、本動作例においては、説明を簡単にするた

め、DFとデータ・バッファの間のSpill/Fillの動作は行わないものとする。

FLの構成要素であるデスティネーション用割付キュー(AQD) 72a及びAMF 3aの当初の内容が第8図に示されるようなものであるとしよう。ここで、循環型のFIFOキューの構成となっているAQD 72aで、キューの
5 先頭からの4エントリが次に発行される命令におけるf1, f2, f3, f4にそれぞれ対応することを示している。AMF 3a及びCMF 3cでは、各々のエントリに上から順に0, 1, 2, ... とアドレスが付けられていて、それぞれs0, s1, s2, ... に対応するものとしている。

10 計算機システムが第8図に示される状態にあるときに、命令1の発行が行われるものとする。

命令1のOpフィールドの内容に基づき命令に含まれるオペレーションの内容がOQに書き込まれ、SMフィールドの内容に基づきAMF及びBaの内容が変更され、その変更の内容がSMQに書き込まれる。この際、命令中のf1, f2, f3, f4の各々はそれぞれフリー・リストから順に取り出されるp26, p34, p42, p51
15 に、置き換えられる。AMFの内容の変更に関しては、アドレス0, 1のエントリには、それぞれf1, f4から置き換えられるp26, p51が書き込まれ、それ以下の部分では、スタックの成長量(2)だけエントリの内容がシフトする(第8図におけるAMFのアドレス0, 1, ... の各エントリの内容がアドレス2, 3, ... の各エントリにそれぞれ書き込まれる)。第9図のOQで書き込み内容を示した
20 4エントリにおいて、リポート・フィールドには、同じ命令1に基づく書き込みが行われるSMQのエントリのアドレス1と識別番号(順にA, B, C, D)が書き込まれている。また、これらのOQの4エントリのディスパッチ・フラグ・フィールドには0が書き込まれている。

f1, f2, f3, f4にそれぞれ対応して新たに割り付けられるDFのアドレスp26,
25 p34, p42, p51の各エントリにおいては、WFフィールドを0とする。

SMQの書き込みが行われるエントリの各オペレーション・ステータス・フィールドには、いずれも対応するオペレーションが存在するので、0が書き込まれている。

5 こうして、計算機システムは第8図の状態から第9図に示されるような状態に至る。

次のサイクルでは、命令1に続いて命令2の発行が行われる。

命令1の場合と同様に、命令2のOpフィールドの内容に基づき命令に含まれるオペレーションの内容がOQに書き込まれ、SMフィールドの内容に基づきAMF及びBaの内容が変更され、その変更の内容がSMQに書き込まれる。この際、命令中のs0, s1は変更前(第9図)のAMFのそれぞれアドレス0, 1のエントリの内容p26, p51に、f1, f2, f3の各々はそれぞれフリー・リストから順に取り出されるp16, p18, p57に、置き換えられる。SMフィールドに負のスタック成長量(-2)しか示されていないので、AMFに関しては、それだけエントリの内容がシフトする(第9図におけるAMFのアドレス2, 3, ...の各エントリの内容がアドレス0, 1, ...の各エントリにそれぞれ書き込まれる)。

15 f1, f2, f3にそれぞれ対応して新たに割り付けられるDFのアドレスp16, p18, p57の各エントリにおいては、WFフィールドを0とする。

さらに、命令2はストア・オペレーションを含むので、その内容が、OQに書き込まれると共に、LSUに送られる。

20 こうして、計算機システムは第10図に示されるような状態に至る。

OQに保持される未実行のオペレーションは、実行可能となったものから、適切な機能ユニットにディスパッチされ、処理される。

第11図は、数サイクルが経過して、命令1以前に発行された命令が全て完了し、命令1に含まれる4つのオペレーション全てが正常終了した時点の計算機システムの状態を示したものである。命令1に対応するSMQのアドレス1

のエントリにおいて、全てのオペレーション・ステータス・フィールドが1となっている。また、この時点のCMF及びB cの内容は、命令1の発行直前（第8図）のAMF及びB aの内容と同じものとなっている（DFとデータ・バッファの間のSpill/Fillの動作が行われれば、命令発行時のB aの値と命令完了時のB cの値は一致しない）。この状態から命令1の完了が次のように実行される。

第11図においてSMQのアドレス1のエントリがその時点のキューの先頭であるので、その内容に基づきCMF及びB cの内容が変更される。すなわち、B cの値に2が加えられ、CMFのアドレス0, 1のエントリにはそれぞれp26, p51が書き込まれ、それ以下の部分では、スタックの成長量(2)だけエントリの内容がシフトする。さらにSMQにおける上記先頭のエントリがキューから除外される。

また、命令1の完了に伴い割り付けを解除すべきDFのエントリのアドレスがFLに登録される。この場合、命令1に含まれるオペレーションを保持していたOQのエントリ群のデスティネーション・フィールドに書き込まれていてCMFへの書き込みが行われないp34, p42がFLの構成要素である命令完了用登録バッファ（EBIC）71aに書き込まれる。（この場合、命令の完了に伴う内容変更によりCMFにおいて保持されなくなるもの、はない。）こうして、計算機システムは第12図に示されるような状態に至る。

第13図は、さらに数サイクルが経過して、命令2に含まれる3つのオペレーションが正常終了し、最後のストア・オペレーションに関してはストア・データとストア・アドレスが共に確定済となった時点の計算機システムの状態を示したものである。EBICに書き込まれたp34, p42がAQDに移されている。

第13図の時点から1サイクルで、命令2の完了が命令1の場合と同様に行われる。この場合、命令2がストア・オペレーションを含むので、そのストアの実行がLSUに依頼される。また、命令の完了に伴い割り付けを解除すべきD

FのエントリのアドレスがFLに登録される。この場合、命令の完了に伴う内容変更によりCMFにおいて保持されなくなるものであるp51, p26と、命令2に含まれるオペレーションを保持していたOQのエントリ群のデスティネーション・フィールドに書き込まれていてCMFへの書き込みが行われないものであるp16, p18, p57がEBICに書き込まれる。こうして、計算機システムは第14図に示されるような状態に至る。

以上で、本実施例の計算機システムにおいて $Y=A*X+B/X$ の計算が完了したことになる。

(4). レジスタ・モード

10 (4-1) レジスタ・モードにおける動作に必要な機能と構成

(3-1)で述べた機能と構成は、一部を除き、レジスタ・モードにおいても必要とされるものである。ここでは、スタック・モードの場合との相違点について説明する。

まず、マッピング・ファイルは、レジスタ・モードにおいては、レジスタ・ベースのスーパースカラ・プロセッサにおけるレジスタ・マッピング・テーブルと同等なものとして用いられる。

従って、マッピング・ファイルのボトム・ポインタ、データ・バッファ、ロード／ストア・ユニット(LSU)のSpill/Fillに関する機能、フリー・リスト(FL)中のスタック・スピル用登録バッファ(EBSS)及びスタック・フィル用割付キュー(AQSF)などは、基本的に、必要ではない。

(4-2) レジスタ・モードにおける動作

ついで、本発明実施例の計算機システムのレジスタ・モードにおける動作を説明する。

本実施例の計算機システムは、レジスタ・モードにおいてもスタック・モードの場合と同様に、基本的に、命令を、①命令フェッチ、②命令発行、③オペレーション実行、④命令完了の4段階で処理する。以下に各段階ごとに動作内容を説明する。

5 ① 命令フェッチ

この段階では、命令フェッチ・ユニットが命令キャッシュから命令を取り出すと共に、次にフェッチする命令のアドレスを決定する。

 ② 命令発行

この段階では、発行する命令のOpフィールドの内容に基づくオペレーション
10 群の発行（オペレーション・キュー（OQ）への書き込み）、SMフィールドの内容に基づく前進マッピング・ファイル（AMF）の内容変更、及びその変更の内容の状態変更キュー（SMQ）への書き込みが行われる。この際、命令中のr0, r1, r2, ... の各々はそれぞれ変更前のAMFのアドレス0, 1, 2, ... のエン
15 トリの内容に、f1, f2, ... の各々はそれぞれフリー・リスト（FL）から順に
取り出されるDFのエントリのアドレスに、置き換えられる。

AMFの内容変更に関しては、次のように行われる。命令のSMフィールド
に示されるAMFのエントリ・アドレスと符号の組の各々について、符号が上記
の要領でそれぞれ適切なDFのエントリ・アドレスに置き換えられ、アドレスが
示されるAMFのエントリに書き込まれる。

20 発行される命令に含まれるオペレーションの内容が書き込まれるOQの各エ
ントリにおいて、リポート・フィールドには、同じ命令に基づく書き込みが行わ
れるSMQのエントリのアドレスと識別番号（順にA, B, C, Dと機械的に付す
ものとする）が書き込まれる。OQのエントリのディスパッチ・フラグ・フィー
ルドには0が書き込まれる。

25 命令の発行に伴い（f1, f2, f3, ... に対応する）新たに割り付けられるD

Fのエントリにおいては、WFフィールドに0が書き込まれる。

ストア・オペレーションの場合には、その内容が、OQに書き込まれると共に、LSUに送られる。

5 OQの書き込みが行われるソース・フィールドに対応するSWFフィールドはまず0とされるが、r0, r1, r2, ... のいずれかから置き換えられたDFのエントリ・アドレスのソース・フィールドへの書き込みの各々に関しては、次サイクルにおいて、そのアドレスのDFのエントリのWFフィールドの内容が読み出され、これが1であれば、OQの対応するSWFフィールドが1に変更されるようになっている。

10 発行される命令に基づき書き込みが行われるSMQのエントリにおいて、対応するオペレーションの存在するオペレーション・ステータス・フィールドには未実行を意味する0が書き込まれ、その他には1が書き込まれる。

③ オペレーション実行

15 OQに保持される未実行のオペレーションは、スタック・モードの場合と同様に、実行可能となったものから、適切な機能ユニットにディスパッチされ、処理されるようになっている。

④ 命令完了

命令の完了は、以下の点を除き、スタック・モードの場合と同様に、プログラム上の順番で行われる。

20 まず、命令の発行の際にAMFに対して行われた内容変更がCMFにおいて再現されるが、それは、レジスタ・モードにおける命令のSMフィールド内部のフォーマット、即ち状態変更キュー(SMQ)の書き込みの形式に即して行われる。

また、命令の完了に伴い、割り付けを解除すべきDFのエントリアドレス
25 がそれぞれFLに登録される。レジスタ・モードにおいては、割り付けを解除す

べきDFのエントリ群のアドレスは次の2つのグループからなる。命令の完了に伴い内容が書き換えられるCMFの各エントリの変更前の内容を読み出したもの、及び、命令に含まれるオペレーションを保持していたOQのエントリ群のデスティネーション・フィールドに書き込まれていてCMFへの書き込みが行われないもの、である。

以上が、本発明実施例の計算機システムのレジスタ・モードにおける動作の概要である。

(4-3) レジスタ・モードにおける動作例

ついで、具体的な動作例について説明する。いま、本実施例の計算機システムで、前述の $Y=(A*X+B/X)*2$ を計算する次の2命令からなるプログラムを実行するものとする。

命令1	Op{load f1, 100[r0]; load f2, 400[r0]; mul f3, f1, f2; load f4, 200[r0]} SM{1:f1, 2:f4, 4:f2, 5:f3}
命令2	Op{div f1, r2, r4; add f2, r5, f1; mul f3, f2, 2; store 500[r0], f3} SM{4:f1, 5:f3}

第15図～第21図は、本実施例の計算機システムにおいて、上記プログラムを処理する際の状態の変化を時系列に示した説明図であり、以下ではこの図をもとに詳細な動作を説明する。第15図～第21図において、DF6、OQ5及びSMQ4の各エントリの構成は、それぞれ第4図、第5図及び第6図のものと同一である。FIFOキューの構成となっている構成要素のキューから外れるエントリなどの書き込み内容が意味あるものとして保持されていないような部分には、斜線が施されている。p##が記入されている箇所は、DFのいずれかのエントリのアドレスが書き込まれているが、本動作例の説明において留意する必要がないことを意味する。

FLの構成要素であるデスティネーション用割付キュー (AQD) 72a 及びAMF 3aの当初の内容が第15図に示されるようなものであるとしよう。ここで、循環型のFIFOキューの構成となっているAQD 72aで、キューの先頭からの4エントリが次に発行される命令におけるf1, f2, f3, f4にそれぞれ対応することを示している。AMF 3a及びCMF 3cでは、各々のエントリに上から順に0, 1, 2, ... とアドレスが付けられていて、それぞれr0, r1, r2, ... に対応するものとしている。

計算機システムが第15図に示される状態にあるときに、命令1の発行が行われるものとする。

命令1のOpフィールドの内容に基づき命令に含まれるオペレーションの内容がOQに書き込まれ、SMフィールドの内容に基づきAMFの内容が変更され、その変更の内容がSMQに書き込まれる。この際、命令中のr0は変更前(第15図)のAMFのアドレス0のエントリの内容p12に、f1, f2, f3, f4の各々はそれぞれフリー・リストから順に取り出されるp26, p34, p42, p51に、置き換えられる。AMFの内容の変更に関しては、アドレス1, 2, 4, 5のエントリに、それぞれf1, f4, f2, f3から置き換えられるp26, p51, p34, p42が書き込まれる。第16図のOQで書き込み内容を示した4エントリにおいて、リポート・フィールドには、同じ命令1に基づく書き込みが行われるSMQのエントリのアドレス1と識別番号(順にA, B, C, D)が書き込まれている。また、これらのOQの4エントリのディスパッチ・フラグ・フィールドには0が書き込まれている。

f1, f2, f3, f4にそれぞれ対応して新たに割り付けられるDFのアドレスp26, p34, p42, p51の各エントリにおいては、WFフィールドを0とする。(r0に対応するDFのアドレスp12のエントリでは、WFフィールドが1でデータ"1000"がすでに書き込まれていたものとしている。)

SMQの書き込みが行われるエントリの各オペレーション・ステータス・フィールドには、いずれも対応するオペレーションが存在するので、0が書き込まれている。

5 こうして、計算機システムは第15図の状態から第16図に示されるような状態に至る。

 次のサイクルでは、命令1に続いて命令2の発行が行われる。

 命令1の場合と同様に、命令2のOpフィールドの内容に基づき命令に含まれるオペレーションの内容がOQに書き込まれ、SMフィールドの内容に基づきAMFの内容が変更され、その変更の内容がSMQに書き込まれる。

10 f1, f2, f3 にそれぞれ対応して新たに割り付けられるDFのアドレス p16, p18, p57 の各エントリにおいては、WFフィールドを0とする。

 さらに、命令2は1つのストア・オペレーションを含むので、その内容が、OQに書き込まれると共に、LSUに送られる。

15 また、前サイクルの命令1の発行において、OQのソース・フィールドに r0 から置き換えられた p12 が書き込まれているので、そのアドレスのDFのエントリのWFフィールドの内容が読み出され、これが1であるので、OQにおいてp12が書き込まれているソース・フィールドに対応するSWFフィールドが1に変更される。

 こうして、計算機システムは第17図に示されるような状態に至る。

20 OQに保持される未実行のオペレーションは、実行可能となったものから、適切な機能ユニットにディスパッチされ、処理される。

 第18図は、数サイクルが経過して、命令1以前に発行された命令が全て完了し、命令1に含まれる4つのオペレーション全てが正常終了した時点の計算機システムの状態を示したものである。命令1に対応するSMQのアドレス 1
25 のエントリにおいて、全てのオペレーション・ステータス・フィールドが1とな

っている。また、この時点のCMFの内容は、命令1の発行直前（第15図）のAMFの内容と同じものとなっている。この状態から命令1の完了が次のように実行される。

第18図においてSMQのアドレス1のエントリがその時点のキューの先頭
5 であるので、その内容に基づきCMFの内容が変更される。すなわち、CMF
のアドレス1, 2, 4, 5のエントリにそれぞれp26, p51, p34, p42が書き込ま
れる。さらにSMQにおける上記先頭のエントリがキューから除外される。

また、命令1の完了に伴い割り付けを解除すべきDFのエントリのアドレス
がFLに登録される。この場合、内容が書き換えられるCMFのアドレス1, 2,
10 4, 5の各エントリの変更前の内容（p02, p10, p24, p60）が読み出され、これ
らがFLの構成要素である命令完了用登録バッファ（EBIC）71aに書き
込まれる。（この場合、命令1に含まれるオペレーションを保持していたOQの
エントリ群のデスティネーション・フィールドに書き込まれていてCMFへの
書き込みが行われないもの、はない。）こうして、計算機システムは第19図に
15 示されるような状態に至る。

第20図は、さらに数サイクルが経過して、命令2に含まれる3つのオペレ
ーションが正常終了し、最後のストア・オペレーションに関してはストア・デー
タとストア・アドレスが共に確定済となった時点の計算機システムの状態を示
したものである。EBICに書き込まれたp02, p10, p24, p60がAQDに移さ
20 れている。

第20図の時点から1サイクルで、命令2の完了が命令1の場合と同様に行
われる。この場合、命令2がストア・オペレーションを含むので、そのストアの
実行がLSUに依頼される。また、命令の完了に伴い割り付けを解除すべきD
FのエントリのアドレスがFLに登録される。この場合、内容が書き換えられ
25 るCMFの各エントリの変更前の内容を読み出したものであるp34, p42と、命

令 2 に含まれるオペレーションを保持していた O Q のエントリ群のデスティネーション・フィールドに書き込まれていて C M F への書き込みが行われないものである p18 が E B I C に書き込まれる。こうして、計算機システムは第 2 1 図に示されるような状態に至る。

- 5 以上で、本実施例の計算機システムにおいて $Y=(A*X+B/X)*2$ の計算が完了したことになる。

(5) 例外回復

ここでは、本発明の計算機システムにおける例外回復について説明する。

- ある命令に含まれるオペレーションの実行において例外事象が発生した場合、
10 その命令の発行時点の状態を回復する必要がある（但し、スタック・モードにおいては、すでに行われたデータ・ファイル D F とデータ・バッファの間の Spill/Fill の動作は取り消さない）。

- そのためには、例外が発生した命令以降に発行された命令を全てキャンセルし、その時点の完了マッピング・ファイル C M F （及びそのボトム・ポインタ B c）の内容を前進マッピング・ファイル A M F （及びそのボトム・ポインタ B a）
15 にコピーし、さらに、状態変更キュー S M Q のキューの先頭のエントリから上記例外が発生した命令に対応するエントリまでの各々の内容に基づき A M F （及び B a）の内容変更を行えばよい。

- ある命令以降に発行された命令を全てキャンセルするには、オペレーション・
20 キュー O Q 及び S M Q においてキャンセルされるべき命令群に対応する範囲のエントリを全てキューから除外し、キャンセルされるべきオペレーションのデスティネーション・レジスタとして割り付けられた D F のエントリのアドレスの各々を F L に戻せばよい。

キャンセルされるべきオペレーションのデスティネーション・レジスタとし

て割り付けられたDFのエントリのアドレスの各々をFLに戻すには基本的に2つの方法がある。即ち、まともに命令完了用登録バッファEBICに順次書き込む方法と、デスティネーション用割付キューAQDのキューの先頭を示すポインタを書き換えるというものである。後者の方法を採用する場合、AQD

5 のキューの先頭を示すポインタ値として書き込まれる可能性がある値をどこかに保持しておく必要がある。それには、(a) SMQにそのためのフィールドを設け、命令の発行ごとに書き込む、(b) 一種の履歴ファイルを設け、条件分岐オペレーションを含む命令の発行ごとに書き込む、というような方法が考えられる。

以上のように、本発明の計算機システムにおいては、例外が発生した命令が

10 発行された時点のAMF（及びBa）の状態を、基本的に、回復することができるので、正確な例外処理が可能である。

（6）その他の実施例

本発明の計算機システムは、上述の実施例に限られるものではなく、細部の構成の異なる様々な実施例が存在する。たとえば、以下のようなものをあげる

15 ことができる。

（6－1）実施例A

本発明の計算機システムは、従来のスタック・ベースあるいはレジスタ・ベースのプロセッサの命令形式に基づくプログラムを命令キャッシュに蓄えておき、実行時に命令発行段階の前段において、（2）で規定した命令形式に準じるよう

20 変換するような構成とすることもできる。

（6－2）実施例B

本発明の計算機システムは、個々の命令の発行の際に、その命令の完了時に割り付けの解除が必要になるDFのエントリのアドレス、あるいはそれを定めるための情報を、状態変更キューあるいは専用のキューにあらかじめ記入して

おくような構成とすることもできる。

(6-3) 実施例C

共にボトム・ポインタを備えるスタック型の前進マッピング・ファイル及び完了マッピング・ファイルと、レジスタ型の前進マッピング・ファイル及び完了マ
5 ッピング・ファイルを具備する構成とし、命令の状態変更フィールドで、スタック型のマッピング・ファイルの内容変更とレジスタ型のマッピング・ファイルの内容変更を共に示すことのできるような計算機システムも実現可能である。

(6-4) 実施例D

レジスタ・ベースのスーパースカラ・アーキテクチャにおいて考えられる様々
10 な変形の多くが、本発明に基づく計算機システムにおいても適用できる。例えば、以下のようなものがあげられる。

- ① 各機能ユニットの入力段にそれぞれリザベーション・ステーションを設けることによってオペレーション・ウィンドウを実現したもの。
- ② オペレーション・キューの後段にスケジューラを設けたもの。
- 15 ③ 整数データ用／浮動小数点データ用、あるいは汎用／マルチメディア用というようにデータ型別にデータ・ファイル、オペレーション・キュー、機能ユニット群、フリー・リストを設けたもの。
- ④ 複数組の前進マッピング・ファイル、完了マッピング・ファイル、オペレーション・キュー、状態変更キュー等を設け、複数のスレッドを並行して処理できる
20 構成としたもの。

産業上の利用可能性

本発明によれば、高性能の計算機システムを実現することができる。また、従来のスタック・ベースあるいはレジスタ・ベースのプロセッサの機械語で記述されたプログラムは、本発明の計算機システムの命令形式に容易に変換できる。

請 求 の 範 囲

1. 各々のエントリにデータが書き込まれるようになっているデータ・ファイルと、

5 各々のエントリに該データ・ファイルのエントリのアドレスが書き込まれるようになっている前進マッピング・ファイル及び完了マッピング・ファイルと、

各々のエントリに該前進マッピング・ファイルの状態変更の内容が書き込まれるようになっている状態変更キューと、

それぞれ該状態変更キューの個々のエントリに対応して設けられたオペレーション群の実行状態を保持する状態保持手段群と、

10 発行済のオペレーションを out-of-order で処理する手段を具備し、

該前進マッピング・ファイルの内容変更、その変更の内容の該状態変更キューへの書き込み、及び該前進マッピング・ファイルの該内容変更と組み合わせさせたオペレーション群の発行を各々 1 サイクルで行うようになっており、

15 発行済のオペレーションの各々は out-of-order で実行され、正常終了すると、これに応じて該状態保持手段群の対応する部分が書き換えられるようになっており、

20 該状態変更キューの先頭のエントリの内容で示される該前進マッピング・ファイルの内容変更と組み合わせさせたオペレーション群に含まれるオペレーションが全て正常終了したことが対応する状態保持手段によって示された後に、該状態変更キューの該先頭のエントリの該内容に基づき該完了マッピング・ファイルの内容変更を行い、該状態変更キューから該先頭のエントリを除外するようになっている計算機システム。

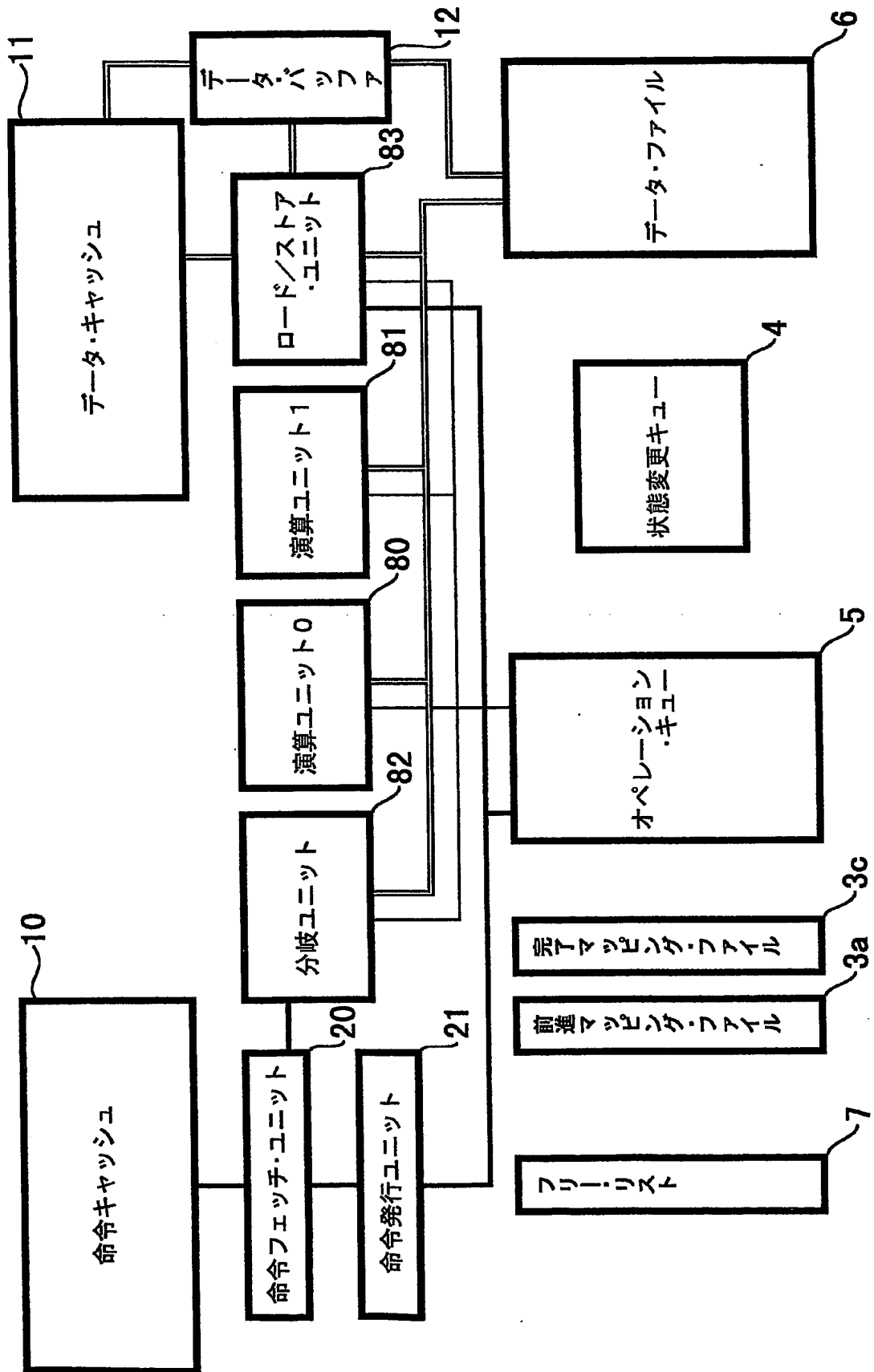
2. 該前進マッピング・ファイル及び該完了マッピング・ファイルがスタックの内容を該スタックの先頭からエントリ・アドレス順に保持するようになってい

る請求項 1 記載の計算機システム。

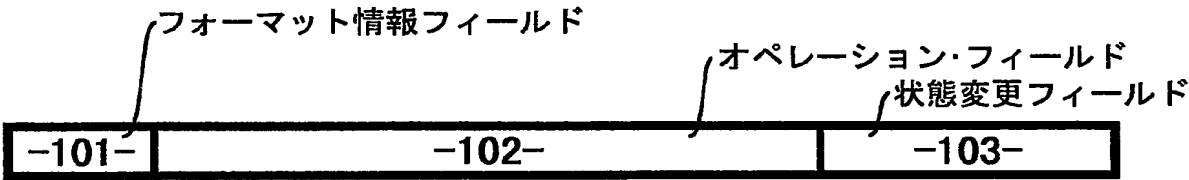
3. 該前進／完了マッピング・ファイルをアクセスするのに、エントリ・アドレスを示すようになっている請求項 1 記載の計算機システム。

1/18

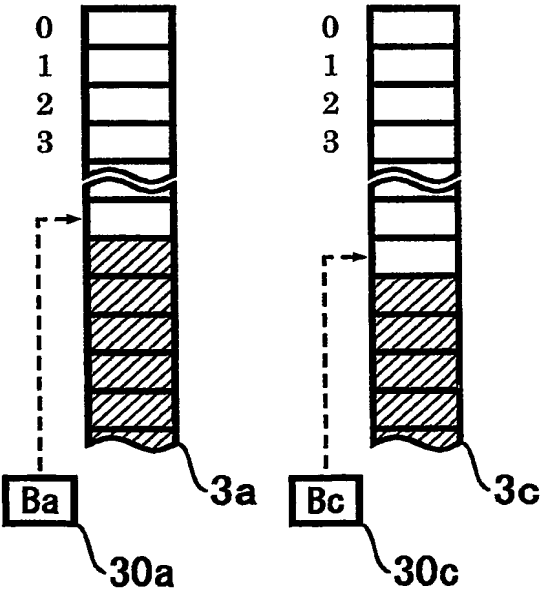
第 1 図



第 2 図

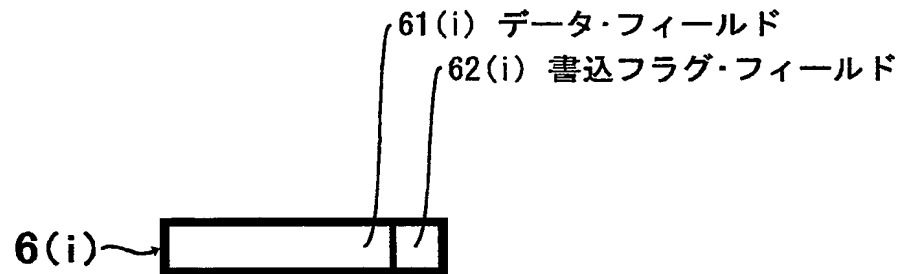


第 3 図

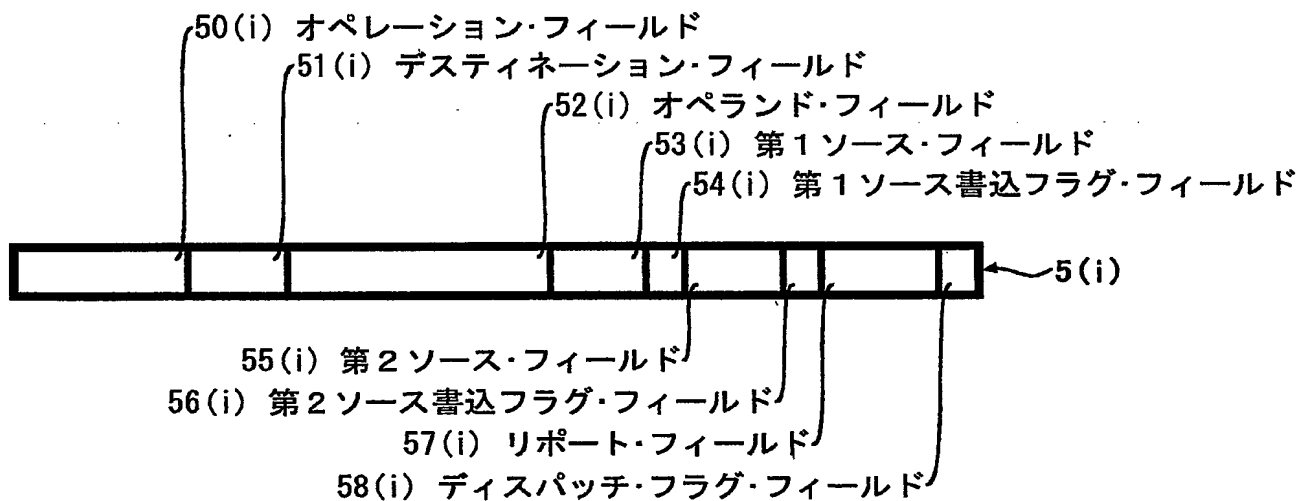


3/18

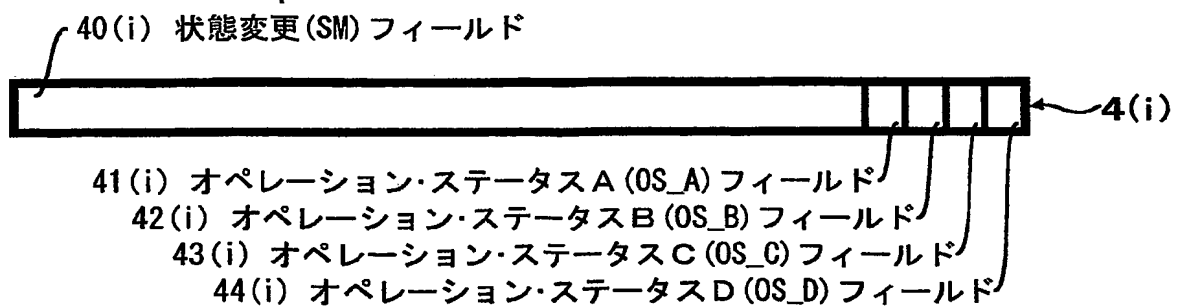
第 4 図



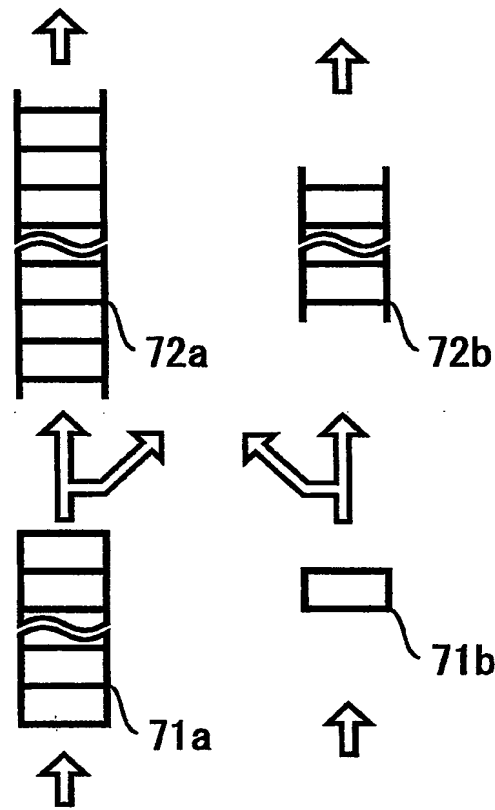
第 5 図



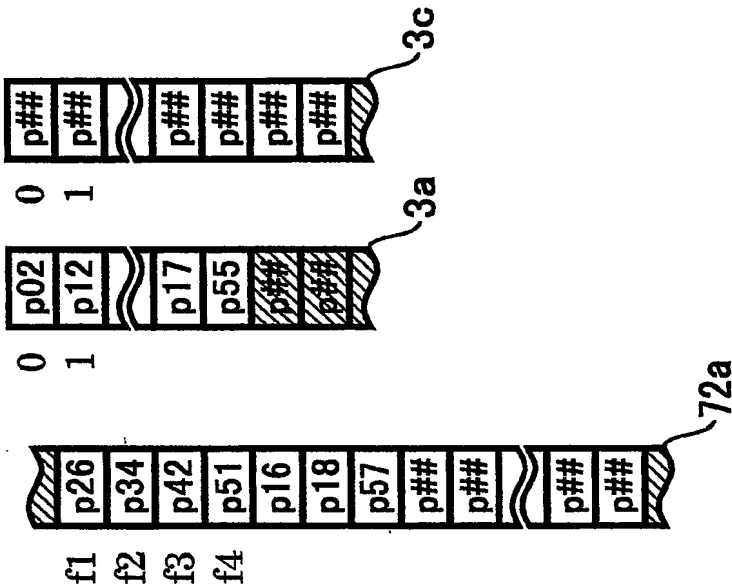
第 6 図



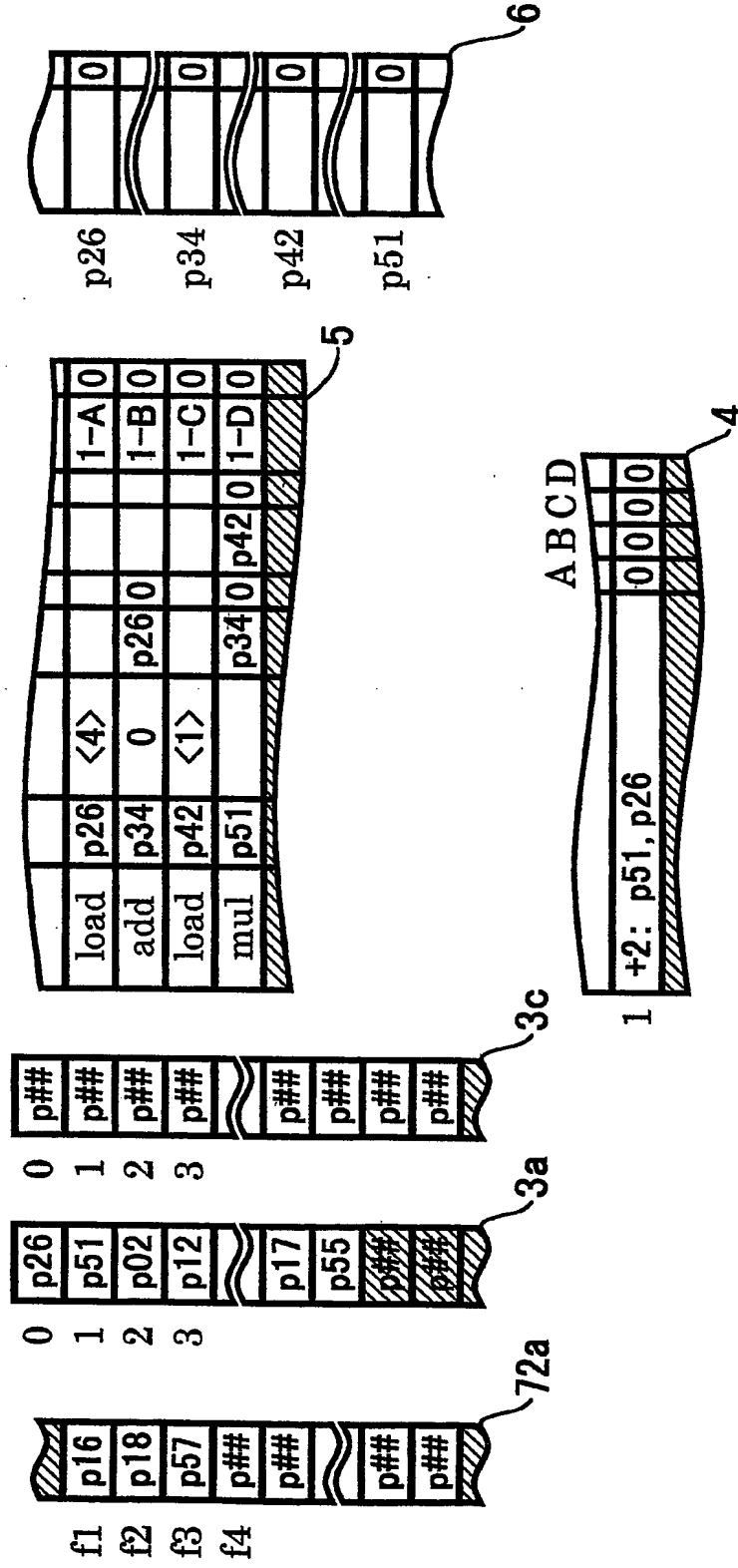
第 7 図



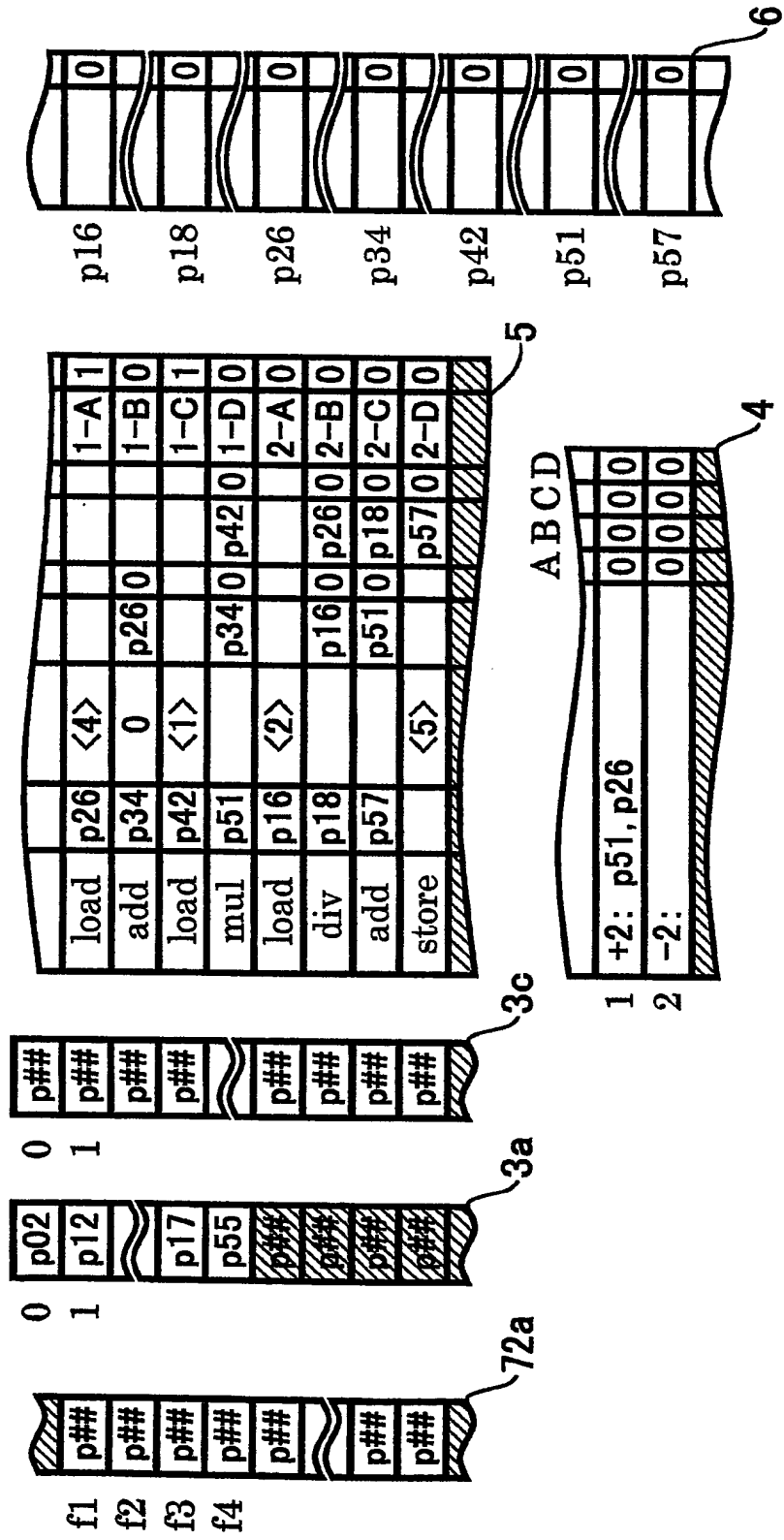
第 8 図



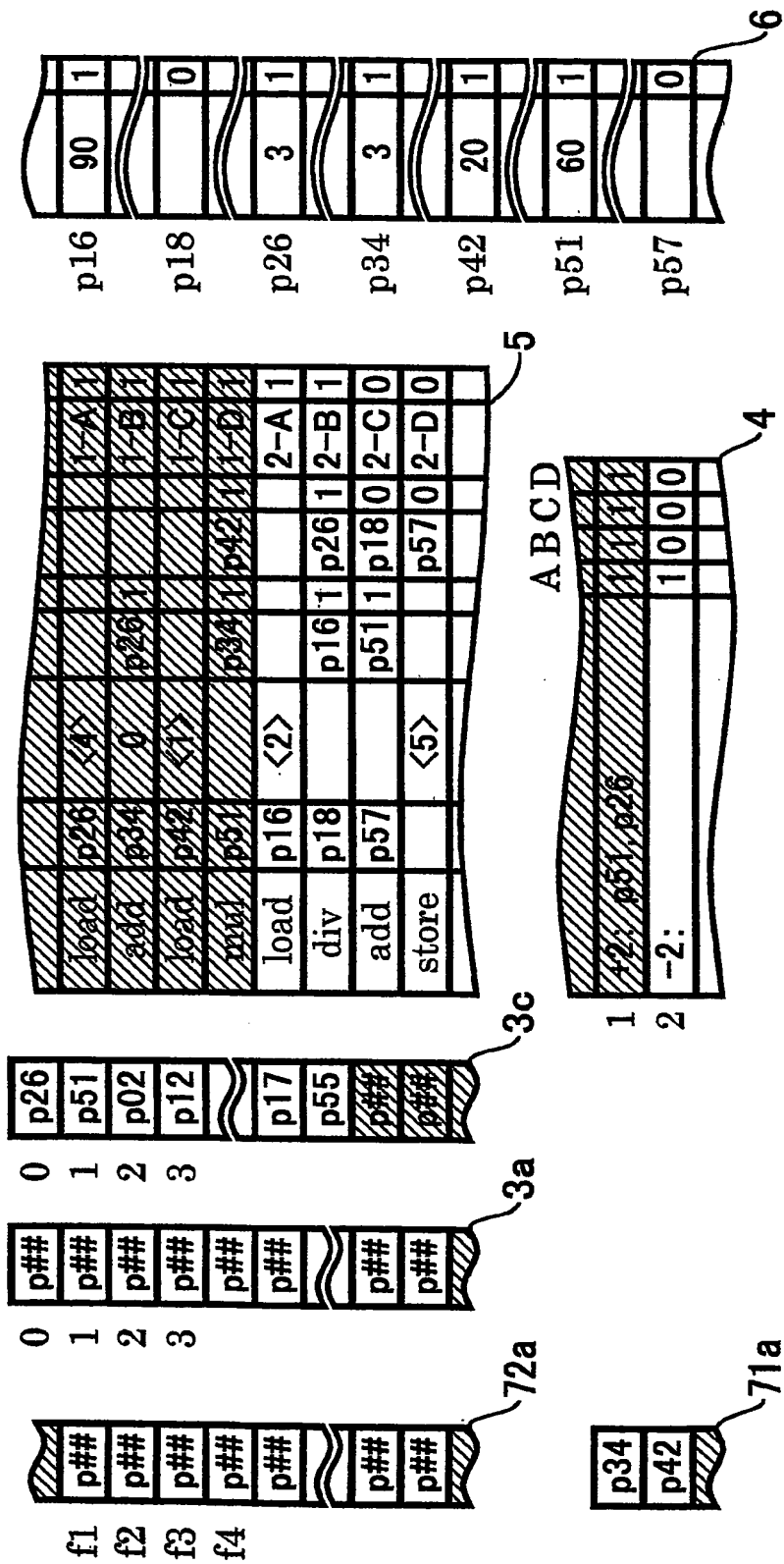
第 9 図



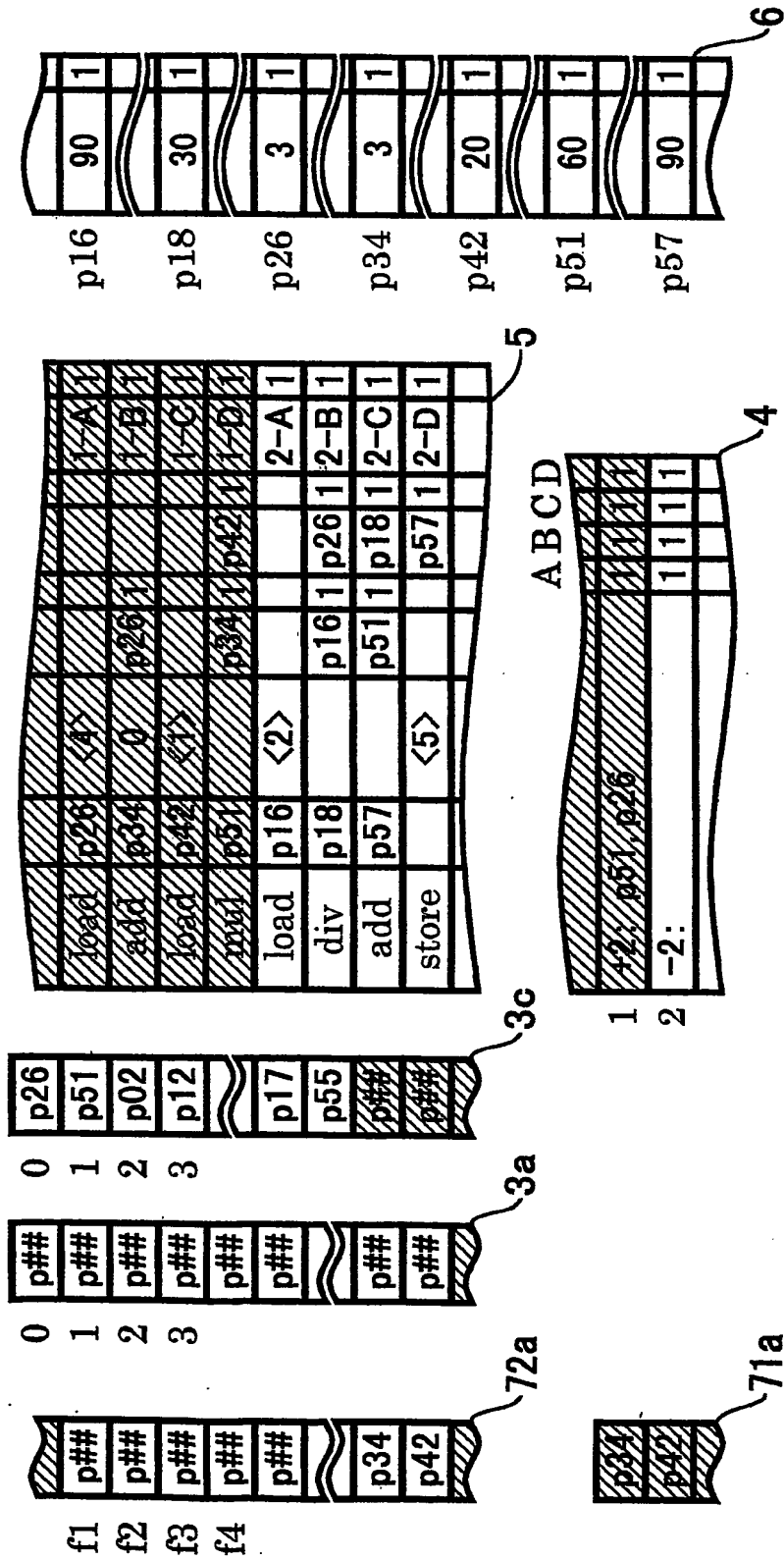
第 10 図



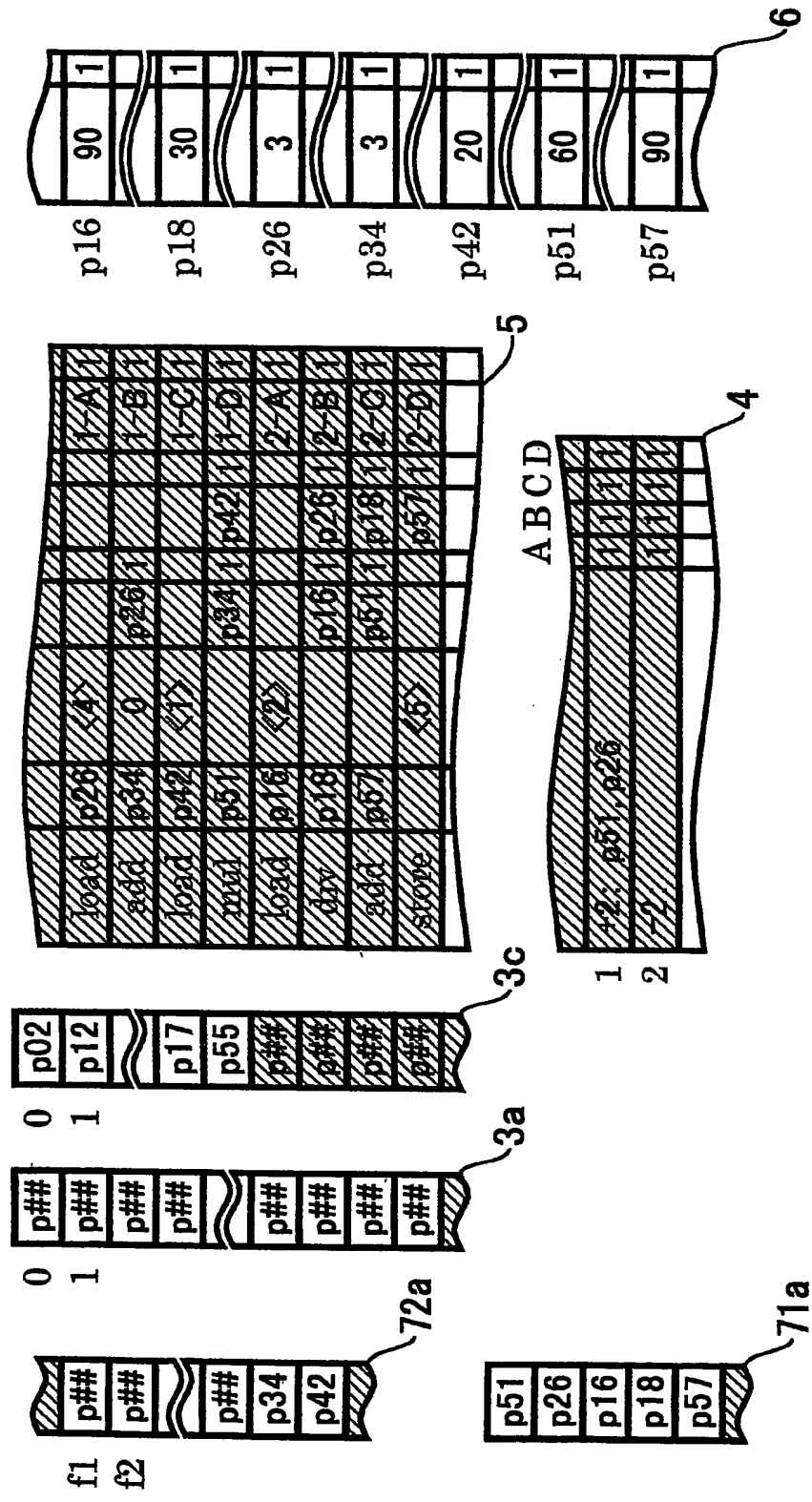
第 1 2 図



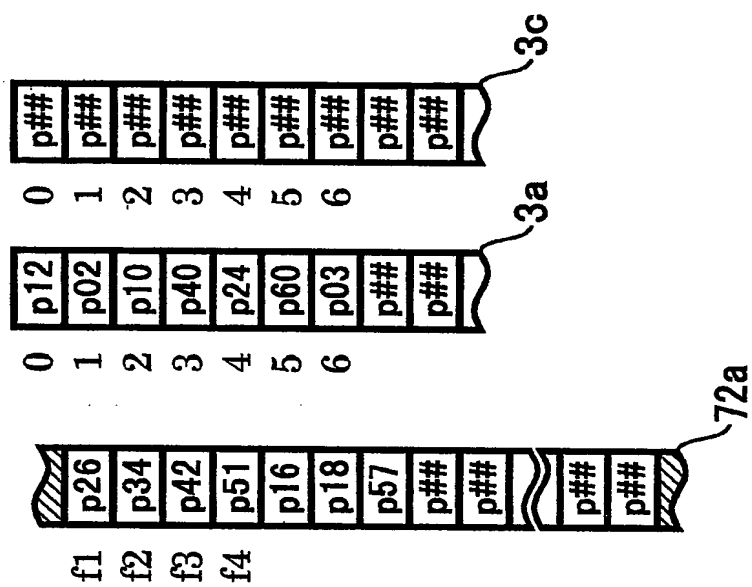
第 1 3 図



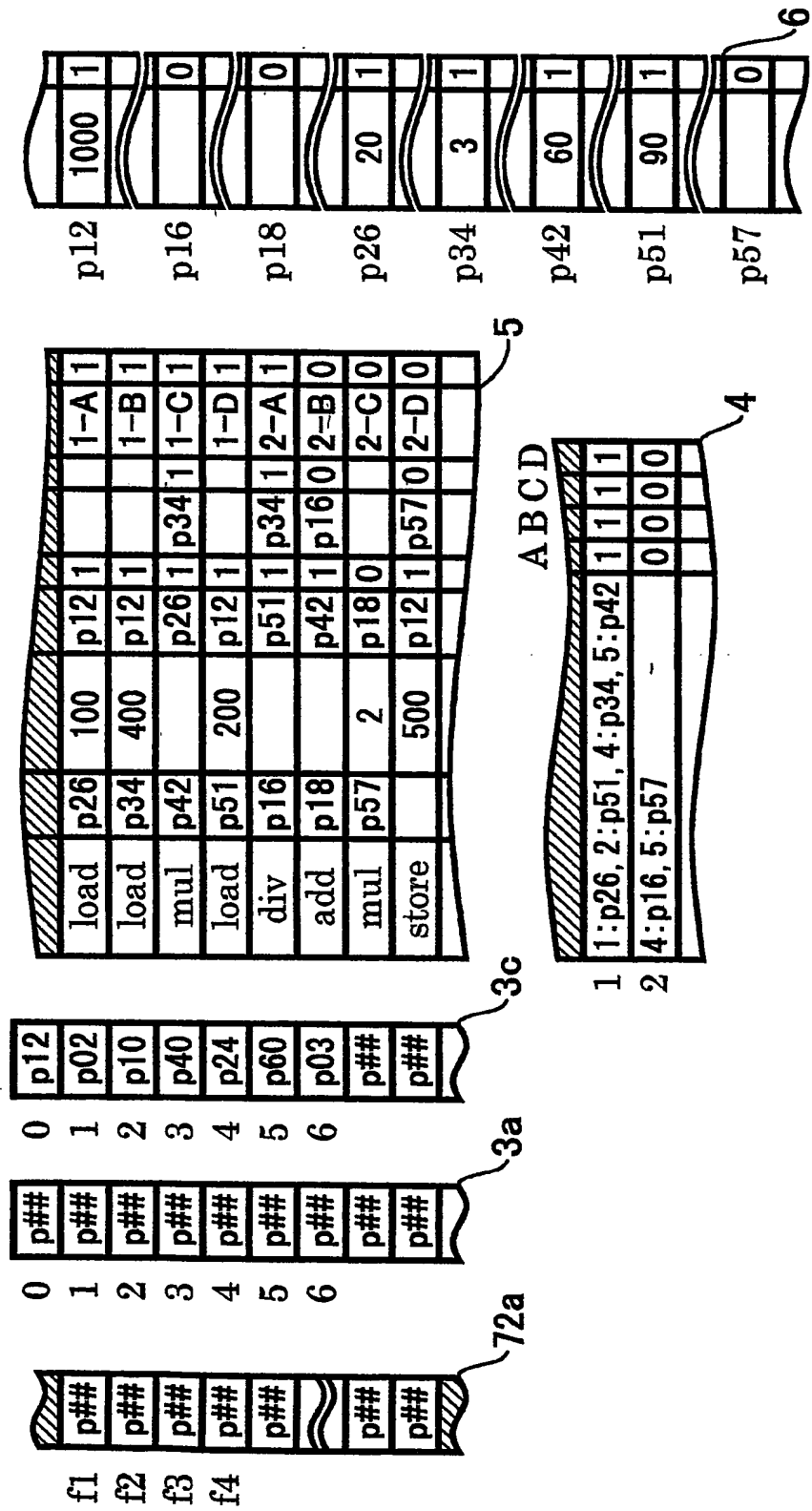
第 1 4 図



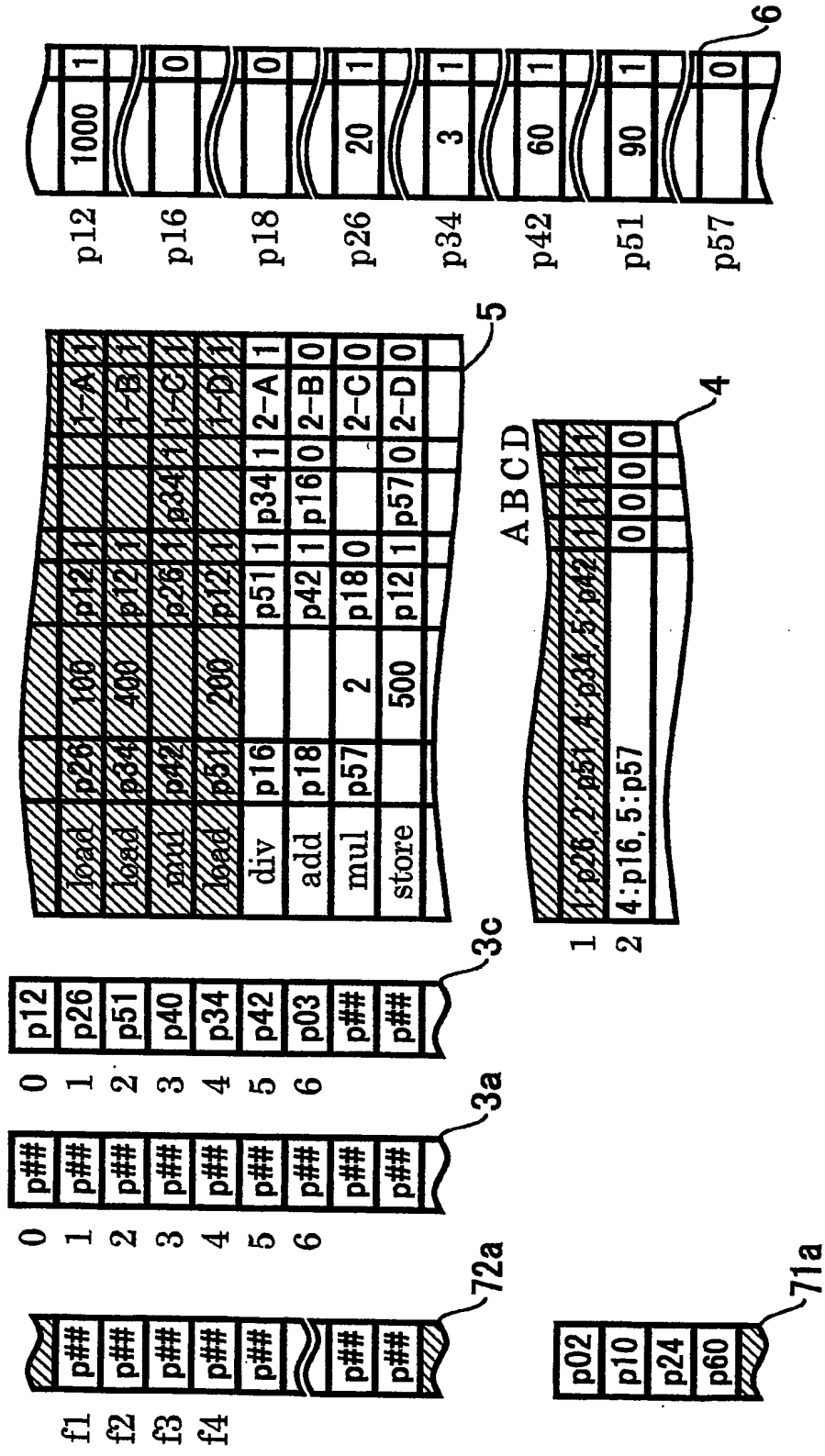
第 1 5 図



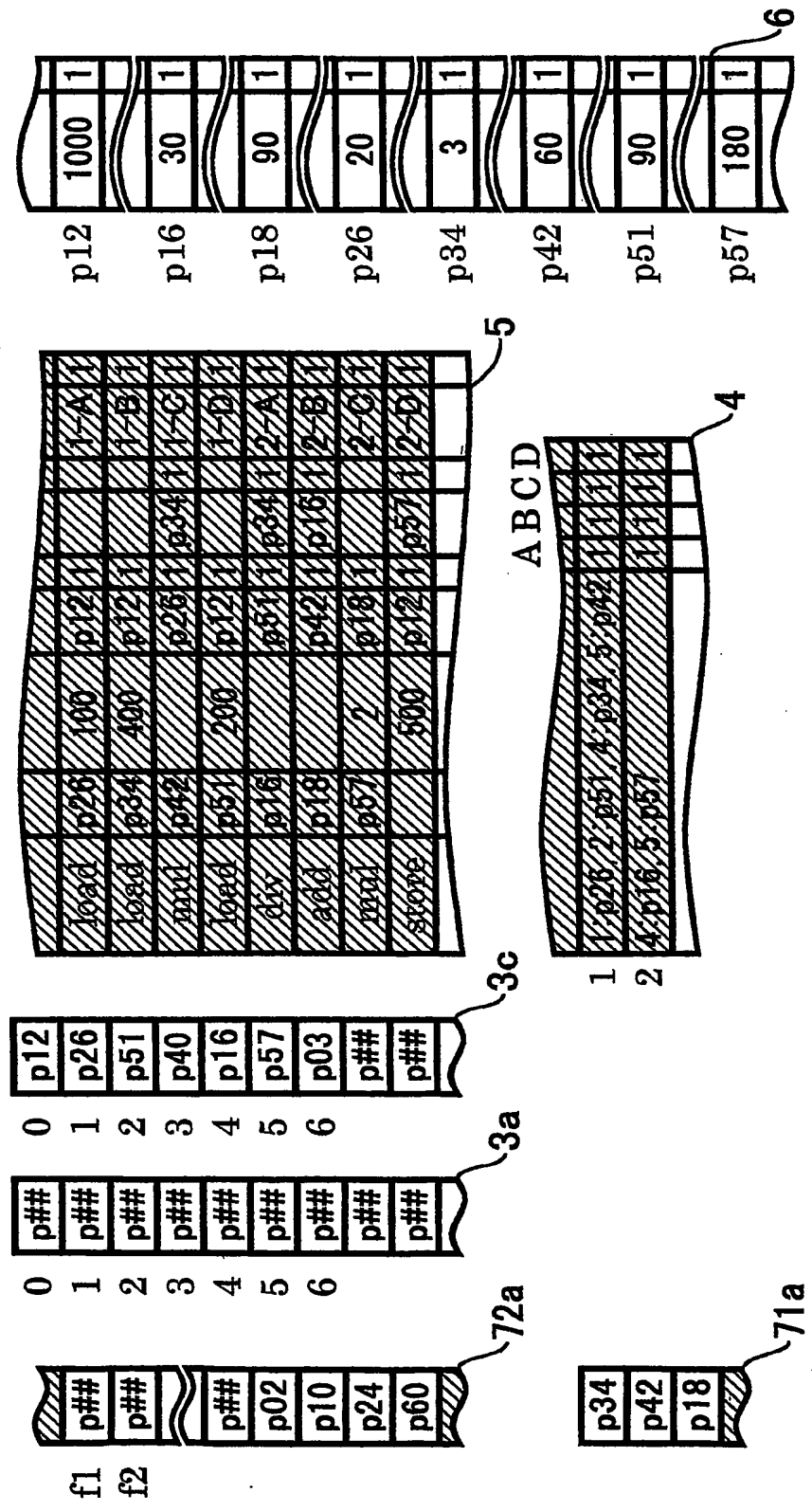
第 18 図



第 1 9 図



第 2 1 図



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP03/10891

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl⁷ G06F9/38

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl⁷ G06F9/38

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Toroku Jitsuyo Shinan Koho	1994-2003
Kokai Jitsuyo Shinan Koho	1971-2003	Jitsuyo Shinan Toroku Koho	1996-2003

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 02/19101 A1 (Hajime SEKI), 07 March, 2002 (07.03.02), Full text; all drawings & EP 1324192 A1	1-3
A	JP 6-28182 A (Matsushita Electric Industrial Co., Ltd.), 04 February, 1994 (04.02.94), Full text; all drawings (Family: none)	1-3

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:
 "A" document defining the general state of the art which is not considered to be of particular relevance
 "E" earlier document but published on or after the international filing date
 "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
 "O" document referring to an oral disclosure, use, exhibition or other means
 "P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
 "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
 "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
 "&" document member of the same patent family

Date of the actual completion of the international search
14 October, 2003 (14.10.03)

Date of mailing of the international search report
28 October, 2003 (28.10.03)

Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl⁷ G06F 9/38

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl⁷ G06F 9/38

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年

日本国公開実用新案公報 1971-2003年

日本国登録実用新案公報 1994-2003年

日本国実用新案登録公報 1996-2003年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	WO 02/19101 A1 (関一) 2002. 03. 07, 全文, 全図 & EP 1324192 A1	1-3
A	JP 6-28182 A (松下電器産業株式会社) 1994. 02. 04, 全文, 全図 (ファミリーなし)	1-3

☐ C欄の続きにも文献が列举されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの

「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)

「O」口頭による開示、使用、展示等に言及する文献

「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

14. 10. 03

国際調査報告の発送日

28.10.03

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

後藤 彰

5B

4226

電話番号 03-3581-1101 内線 3545